EXPLOITING WIRELESS BROADCASTING NATURE FOR HIGH-THROUGHPUT 802.11 MESH NETWORKS

by

JIAN ZHANG

A Dissertation submitted to the

Graduate School-New Brunswick

Rutgers, the State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Graduate Program in

Electrical and Computer Engineering

written under the direction of

Professor Ivan Marsic

and approved by

____

____

____

____

New Brunswick, New Jersey

October, 2008

ABSTRACT OF THE DISSERTATION

Exploiting Wireless Broadcasting Nature for High-Throughput 802.11 Mesh Networks

by Jian Zhang

Dissertation Director:

Professor Ivan Marsic

Wireless mesh networking (WMN) has seen great research and commercial interests recently. It is considered a promising technology for implementing wireless community networks. Although the ubiquitous and low-cost Wi-Fi devices make IEEE 802.11 a prevailing choice for wireless mesh networks, the current IEEE 802.11 protocols cannot achieve full utilization of network capacity in wireless mesh networks. When dealing with multi-hop transmissions, IEEE 802.11 MAC presents low efficiency in coordinating concurrent transmissions, ineffectiveness in avoiding interference. In this dissertation, we demonstrate the factors that lead to low capacity in 802.11 mesh netowrks, and we survey the existing work on improving the network throughput. A large class of previous work focuses on coping with the interference, i.e., curbing the negative effect of broadcast nature of wireless signaling. In contrast, another set of work introduced here tries to exploit the wireless broadcasting, and it exhibits promising potential for large throughput gain in mesh networks.

We propose three methods that further take advantage of wireless broadcasting. They all aim at achieving high throughput in WMNs, while exploiting the broadcasting nature in different aspects.

The first work passively makes use of wireless broadcasting in that it collects wireless link information through mere listening. Specifically, we present a non-intrusive method to model and estimate 802.11 link bandwidth based on radio signal-to-noise ratio (SNR).

In a more active way, the second method enables mesh senders to probes their receivers for their MAC statuses so that their following transmissions can be more efficient. Owing to wireless broadcasting, multiple receivers are probed simultaneously with one single probing and their diversity/correlations are speculated. With the diversity information, we propose a smart scheduling strategy. We show that such diversity information can greatly improve throughput of mesh senders.

Furthermore, the third method proactively uses the packet redundancy caused by wireless transmissions in local area of a network. Such redundancy are usually ignored or even avoided on purpose by traditional protocols. In contrast, we show that it provides an abundant repository of packets for performing network coding. We propose a new protocol called BEND, which enables each potential forwarder to proactively mix/encode the packets that either are intended to or are overheard by this node. This proactive mixing significantly increases the coding opportunities in the network, leading to high capacity gain.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

There is growing interest in providing ubiquitous wireless broadband access within metropolitan, suburban or rural areas. Muniwireless' list of municipal WiFi networks shows that at the end of 2006, there were 312 cities and counties in the US with their wireless networks running, or in the deployment or planning phase — that's triple the number from early 2005 [66]. Among various wireless technologies used to build region or citywide networks, Wi-Fi (IEEE 802.11) is the most dominant. Such preference of Wi-Fi is due to its high data rates, the license-free spectrum, and the low cost and the widespread of 802.11 devices. In particular, Wi-Fi technology offers higher data transmission rates, hence larger network capacity than the cellular networks, another candidate solution for anytime anywhere data service. Most of the currently available 802.11 devices can achieve a data rate up to 54Mbps. The rate is supposed to be further boosted, e.g., by equipping MIMO (multiple-input multiple-output) antennas.

Traditional Wi-Fi network, i.e. so called wireless LAN (WLAN), is operating in infrastructure mode, where all the hosts in an infrastructure basic service set (BSS) must be in direct communication range of a specific access point (AP). Although its service range can be extended by adding more BSS's, the scaling could be costly. Establishing and maintaining such centralized infrastructure for each BSS alone is difficult. In addition, all APs are supposed to have wired network backhaul connection. This impedes the deployment of wireless network in areas without wired connections.

Considered a promising technology substitute for WLAN in building large-scale wireless networks, wireless mesh network (WMN) has seen great research and commercial interests recently. Contrast to WLAN, WMN is infrastructure-less and has relay function. Due to this, WMNs present features like extended coverage and low deployment cost. However, these features of WSNs do not come free. Also due to their multi-hop decentralized architecture, WMNs present greater complexity relative to conventional single-hop wireless networks, demanding more complex medium access control (MAC) to deal with mutual interferences, a need for efficient and robust message forwarding in the presence of unreliable links, a need of cross-layer design for higher network capacity, and so on.

Although the ubiquitous and low-cost 802.11 devices make IEEE 802.11 protocols a natural choice for implementing WMN, the current 802.11 scheme cannot achieve full utilization of network capacity. Dealing with multi-hop transmissions in mesh networks, IEEE 802.11 MAC presents low efficiency when coordinating concurrent transmissions, ineffectiveness when avoiding interference, and thus low network capacity.

It motivates researchers to seek for new techniques in high-performance WMNs. Most of the research has focused on coping with the interference, i.e., curbing the negative effect of broadcast nature of wireless signaling. Recently, there has been a realization that such wireless broadcast nature can provide unexpected performance gain if well utilized. In this dissertation, we describe some attempts we have made to achieve high throughput performance in WMN by exploiting the wireless broadcasting nature.

1.1 Main contribution

We focus our work on taking the versatile and valuable advantages of wireless broadcasting on implementing high-throughput wireless networks. The three approaches provided in the dissertation serve the above goal, whereas exploiting the broadcasting nature in different aspects.

The following paragraphs present the key ideas of our methods with increasing aggressiveness and sophistication.

1. **Passive listening --** ***wireless link bandwidth estimation.*** When “logically” replicated for wireless networks, traditional measurement methods designed for wired networks introduce high overhead by sending probes. The wireless broadcasting nature makes it possible for a station to speculate the link quality with another station by listening instead of probing. In Chapter 5, We present a non-intrusive method to estimate 802.11 link bandwidth based on radio signal-to-noise ratio (SNR), which is collected by simply passive listening.
2. **Active probing --** ***the channel-state-based MRTS technique.*** Besides the link conditions, it is also critical for a mesh sender to detect the MAC statuses of its neighboring receivers in order to ensure the success of its following transmission. Owing to the broadcasting feature, the MAC statuses of multiple receivers and their correlations can be derived simultaneously by one single probing. In Chapter 3, we propose a smart scheduling strategy combined with such probing scheme to improve throughput of mesh senders.
3. **Proactive mixing --** ***an opportunistic forwarding scheme for network coding.***

Wireless broadcasting results in multiple copies of a packet on different nodes in the network. Traditional protocols usually ignore or even attempt to avoid such redundancy. In contrast, we realize that it provides an abundant repository for performing network coding. In Chapter 4, we present a new protocol called BEND, which enables each potential forwarder to proactively mix/encode the packets that either are intended to or are overheard by this node. This proactive mixing could significantly increase the coding opportunities in the network, leading to high capacity gain.

To evaluate the above methods, we use a mix of theoretical analysis, simulations, and experimental results. In particular, we compare the our protocols with original 802.11 and other existing work through analysis and ns-2 simulations. The performance metrics include gateway throughput, network capacity and fairness. For the link bandwidth estimation, due to the lack of accurate physical link modeling and simulation among the existing tools, we verify our model by real data collected through the indoor/outdoor experiments.

1.2 Organization

The remainder of the dissertation is organized as follows. Chapter 2 provides background and reviews. We first present the applications, characteristics, and technical challenges of wireless mesh networks. Then, some existing work to improve performance in WMNs is introduced. Chapter 3 describes our channel-state-based MRTS technique. In Chapter 4, we introduce network coding in multi-hop wireless networks. The idea and the design of BEND for proactive mixing are given. Chapter 5 presents the theory, the approaches, and the experiments for 802.11 link bandwidth estimation.

Chapter 2

Background & Overview

2.1 Wireless multi-hop networks

In WLAN, all the hosts must communicate directly with a centralized station, referred to as Access Point (AP). That is, all the hosts in this infrastructure BSS must be within the transmission range of the specified AP. However, the coverage of an 802.11 AP is limited. For example, operating in open environment with no physical obstructions with 1Mbps modulation rate, an Orinoco 802.11b wireless PC card can cover 1750 ft. while keeping the bit error rate below 10^{-5}. The range reduces to 525 ft. for a modulation rate of 11Mbps. For each single modulation, the bit error rate increases exponentially as the distance between a host and an AP grows. With such single-hop transmission limit, multiple infrastructure BSSs and APs are required to build a wireless network for a large community. The centralized topology of each BSS makes it very time and cost consuming to maintain a large network.

Contrast to access-point-based wireless LANs, wireless mesh networks support multi-hop connections. Multi-hop communication extends the coverage of traditional WLAN and greatly enhances the flexibility in applications. The nodes beyond each other's transmission range can still communicate through message forwarding provided by the intermediate nodes. The communications between any two nodes in the community do not necessarily need or involve a centralized station like AP in WLAN.

To establish the multi-hop connections, each distributed node in a network coordinates with others to find and later maintain the routes automatically. For a new node joining the network, it first searches for neighbor nodes and through them finds routes to other nodes in the network. This is achieved without or with minimal human intervention. Such feature is referred to as self-organizing/self-configuration. Self-organizing is critical for deploying a large-scale wireless network with lost cost.

Moreover, the multi-hop connections alleviate the geographic demands for wired connections to Internet. In a wireless mesh network, a node with wired/wireless Internet connection, called mesh gateway or Internet gateway, is responsible for delivering data between Internet and other nodes in the mesh network. Due to extended coverage by multi-hop connections, mesh gateways in WMN can be sparse. Each mesh gateway serves a portion of mesh hosts including some stations located more than one-hop away from any wired connections.

From above, wireless multi-hop technology provides a low-cost solution for deploying a large wireless community network in un-wired or under-wired areas.

2.2 Characteristics of WMNs and their current developments

Mobile ad-hoc networks (MANET) and wireless mesh networks (WMN) are two most typical types of wireless multi-hop networks. The research on MANETs was originally motivated by mobile application scenarios, such as battlefield and fire fighting. High mobility is presumed when designing MANETs; thus, the network's wireless topology may change rapidly and unpredictably [40]. Unlike MANET, the study of WMNs focuses on commercial implementation and applications. It emerges as a multi-hop solution for civilian applications, such as community and office networking. In contrast to the high mobility in MANETs, mesh networks are comprised of stationary nodes or nodes with low mobility. For example, a node in Roofnet, a mesh networking test bed in MIT, consists of a PC, an 802.11 radio and a roof-mounted antenna [68]. The architecture of a WMN is not necessarily flat or “ad hoc”. Some fixed stations (e.g., roof-mounted and pole-mounted) can form the backbone in a WMN, introducing some level of hierarchy. Although the connectivity among other mobile nodes could still be ad hoc, the effect of mobility on the network topology is relatively low. Thus in WMNs, the power limit and the mobility are not as important design concerns as in MANETs.

Like the AP in WLAN, there is a type of mesh stations in WMN, called mesh gateway, which deliver data packets between mesh stations and the outer world, e.g., Internet. A mesh node could be multiple hops away from its corresponding mesh gateway. This gateway scenario is one of the most typical scenarios for WMN researches.

Some implemented WMN test beds include Roofnet by MIT [68], Champaign-Urbana Community Wireless Network [67], Seattle Wireless [69] and Microsoft Research’s Mesh network test bed [70]. Commercial WMN solutions are provided by Motorola, Cisco, Tropos Networks, BelAir, Nortel and so on. In addition, three Working Groups of IEEE project 802 are working on mesh network standards. The IEEE 802.11s develops amendment for extended service set (ESS) mesh networking; The IEEE 802.15.5 TG plans to provide wireless mesh topologies for wireless personal area networks (WPAN) devices; and the IEEE 802.16j works on wireless relay network solution for 802.16-based devices.

In the existing deployments of mesh networks, IEEE 802.11 protocols are prevailing. The low cost and the ubiquity of 802.11 devices have made 802.11 the first choice for WMNs. Our work also focuses on 802.11-based mesh networks.

2.3 The limited capacity of 802.11-based WMN

Unlike infrastructure WLAN, WMN has its own scaling problem. The capacity of a WMN becomes scarce as the density and the size of the network increase. The multi-hop transmissions complicate the medium access and intensify the contentions in the network, leading to increasing packet losses and thus throughput degradation. WMN undergoes not only inter-flow interferences but also intra-flow interferences. Hence, the throughput of a flow degrades as the number of hops (path distance) increases.

Our work in this dissertation focuses on improving throughput and capacity in WMN. Various enhancements and new technologies have been proposed for this problem. Among them, a great amount of methods focus on enhancing current 802.11 MAC to better support multi-hop transmissions. Through either smarter sensing or adding information exchanges, these protocols coordinate medium accesses and manage their interference more efficiently so that more concurrent transmissions can be accommodated in the network. We survey a class of such methods in section 2.4. In section 2.5, we introduce another type of approaches that use new routing metrics for path finding in WMNs. In sections 2.6, 2.7 and 2.8, we describe some novel technologies that exploit the wireless broadcasting nature and the spatial diversity in the network. These technologies are closely related to our proposed methods. Other work is briefly given in section 2.9.

2.4 The enhancements on 802.11 MAC

The 802.11 MAC regulates the medium access behaviors of users. The wireless channel is shared by multiple users in a WMN, and the transmission of one user may interfere with other ongoing receptions. Therefore, a proper medium access control is critical to the performance of the whole network. The 802.11 standards primarily aim at fulfilling needs of WLANs, i.e., the single-hop wireless networks. Unfortunately, when deployed in multi-hop mesh networks, the regular IEEE 802.11 MAC is limited in its ability to provide high spatial reuse/parallelism and to handle interference scenarios compounded by multi-hop traffic. These lead to low network capacity.

Therefore, some 802.11 MAC enhancements that handle intra-/inter-flow interferences are proposed in order to accommodate more concurrent transmissions, and at the same time, to maintain the required channel quality for each transmission.

2.4.1 Basic carrier sensing scheme of 802.11

The IEEE 802.11 MAC provides two collision avoidance (CA) mechanisms, the mandatory basic CSMA/CA and the optional virtual carrier sensing scheme with RTS/CTS [23]. Under the basic scheme, a station refrains from medium access if it senses any ongoing transmission on the wireless channel. The mechanism to determine whether or not the channel is busy is called clear channel assessment (CCA). A prevalent CCA mode is known as carrier sense with energy detection. That is, the CCA decision is based on whether the energy of a detectable 802.11 signal exceeds a threshold, called *carrier sense threshold*. Given a carrier sense threshold, the corresponding *carrier sense range* is defined as the minimum distance allowed between two concurrent transmitters [56]. On the one hand, it may be true that the smaller the carrier sense range (or the higher the carrier sense threshold), the better the spatial reuse and the higher the efficiency. On the other hand, the interference at a receiver can also increase as the carrier sense range becomes smaller, i.e., as concurrent transmitters get closer, which may impair the effectiveness of the collision avoidance mechanism. An interference model has been developed to describe the relationship among the transmission power, the carrier sense threshold and the aggregate throughput. By such a model the optimal carrier sense threshold is specified to maximize the aggregate throughput for a regular topology, as described next.

2.4.2 Interference model and static carrier sensing

In [56][33], the worse case interference and signal-interference-noise ratio (SINR)

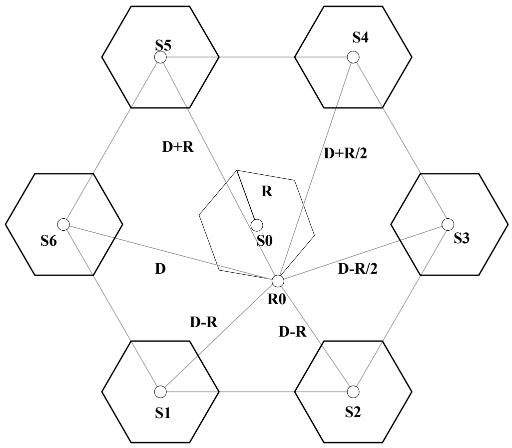


Figure 1 Worst-case interference scenario

at a receiver station is derived.

We denote the carrier sense threshold by T_{cs}, the corresponding carrier sense range by D, the transmission power by P_{tx}, and the transmission range by R. When a sender S_0 is transmitting, a concurrent transmitter must be at least a distance D away from S_0. Therefore, in the worst case there can be a total of 6 interferers distributed on the circle centered at the sender with radius D. This can be approximated by the Honey-grid model [21] as in Figure 1. As illustrated in the figure, the worst case interference occurs when the distances between the receiver R_0 and the six interferers approximately equal $D-R$, $D-R$, $D-R/2$, $D+R/2$, $D+R$, and D, respectively.

It is showed in [33] that the network capacity based on the above model can be defined as a function of P_{tx}/T_{cs}. Therefore, the highest aggregate throughput can be achieved by adjusting either the transmission power P_{tx} or the carrier sense threshold T_{cs}, or both. Some approaches use the above analytical model to determine an invariant optimal value of the carrier sense threshold for all the stations in the network given a fixed transmission power. In practice, however, it is not typical that all of the receivers in a network will experience the worse-case interference. Therefore, instead of holding the carrier sense threshold or transmission power of all nodes constant all the time, some methods are proposed to adjust these parameters dynamically. These dynamic control methods are usually combined with the virtual carrier-sensing scheme as described in the following section.

2.4.3 Virtual carrier sensing of 802.11

As a complement of the basic collision avoidance scheme, virtual carrier sensing [7] is dedicated to solving the collision problem due to hidden stations [51]. The idea is to reserve the wireless channel by preceding the data frame transmission with an RTS/CTS handshake. The neighboring stations that receive the RTS/CTS frames are blocked from transmitting for a period of time specified in the frames. This is done by setting the *network allocation vector* (NAV) of an overhearing node's MAC agent.

In the original design, the blocking area is decided by the transmission range of the RTS/CTS. It assumes that the stations are able to interfere with the upcoming

DATA/ACK frames only if they can receive RTS/CTS, i.e., that the transmission range of control frames equals the interference range. However, there commonly exists a disparity between the RTS/CTS transmission range and the interference range. Instead, it may result in one of the two opposite situations, i.e., either the failure of collision avoidance or unnecessary false blocking, depending on which range is larger.

The interference range, D_I is defined as the shortest distance between the receiver and a interferer so that the SINR on the receiver is right above *capture threshold*, denoted by T_{cap}, when the sender and interferer transmission power levels for DATA frames are P_{tx} and P_{inf}, respectively, i.e., satisfying

$$SINR = \frac{P_{tx}}{r^{\theta}} \bigg/ \frac{P_{\inf}}{D_I} \geq T_{cap} \qquad (1)$$

This shows that the interference range is not a fixed value in that it changes with the actual distance r between the transmitter and the receiver, and with the capture threshold T_{cap} which is decided by the modulation scheme (and thus data rate) used. Thus, it is common that the CTS transmission range does not necessarily match the current interference range. When the transmission range of CTS is smaller than the interference range, the CTS frame cannot be deciphered correctly by all potential interferers, leading to collisions, referred to as ineffectiveness of collision avoidance. On the other hand, a CTS with an excessively large transmission range may cause low spatial reuse, especially in wireless multi-hop networks, referred to as inefficiency.

An example shown in Figure 2(a) assumes that all nodes transmit RTS/CTS/DATA/ACK frames with the same power and modulation scheme. Although node *X* may sense node *R*'s transmission since it is within *R*'s carrier sense range, it cannot decode the CTS frame since it is outside of the transmission range of CTS of node *R*. Therefore, although node *X* will stay silent for the period of this CTS transmission, it may still transmit during the DATA frame from *S* to *R* since it failed to set its NAV based on the CTS frame. This may result in a DATA frame collision since node *X* is within the interference range of receiver *R*. This is the so-called hidden station problem, which still cannot be avoided by the original RTS/CTS scheme.

In order to avoid such collisions, some researchers have proposed to extend the transmission range of RTS/CTS by increasing the RTS/CTS transmission power. For example in [19], the RTS and CTS are sent at the highest power level, and the data and ACK at a lower power level. However, it turns out that the above collision problem cannot be well solved by such a strategy. The reason is that by enlarging the CTS

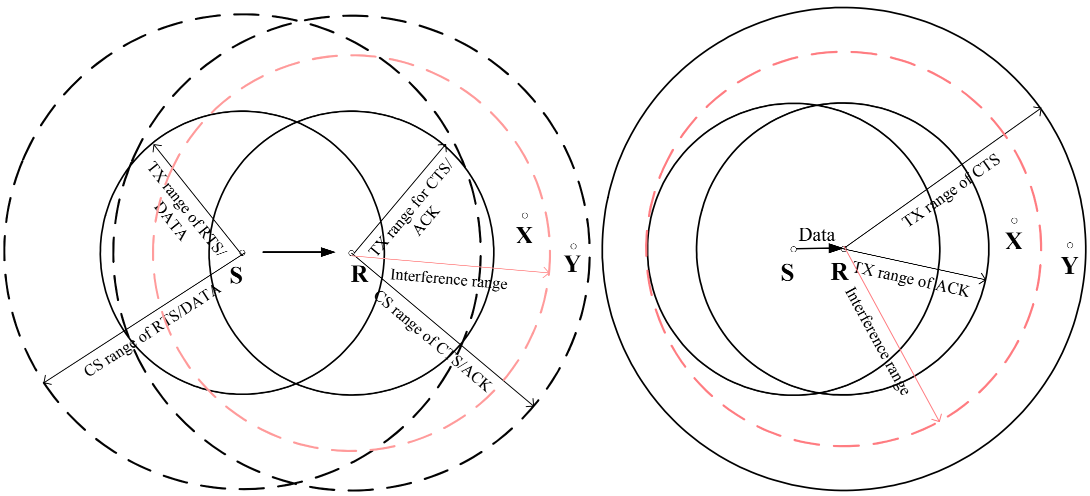


Figure 2 (a) Ineffectiveness of collision avoidance and (b) inefficiency of spatial reuse transmission range of receiver R to defer more potential interferers, at the same time we also increase the interference of RTS/CTS frames at the neighboring nodes due to the higher transmission power, i.e., the interference range is also increased due to a larger P_{inf}. In addition, an excessively large transmission range of CTS may lead to the other extreme situation, i.e., inefficiency. As shown in Figure 2 (b), node Y is unnecessarily blocked although its transmission would not interfere with the data reception of R (because it is beyond its interference range).

Thus, some dynamic control methods are proposed to improve the spatial reuse/efficiency without impairing the effectiveness of collision avoidance. They usually use RTS/CTS frames to exchange power and interference information.

2.4.4 Soft blocking and dynamic power control

The idea, here referred to as "soft blocking" [9] [41], is to conditionally set the NAV of every node that overhears a CTS frame. Assume transmission range of RTS/CTS frames is sufficiently large, as in Figure 2(b). To achieve high efficiency, some node, say node Y, may choose not to set its NAV when overhearing a CTS if it can tell its transmission will not interfere with the reception at receiver R. Node Y decides this by using the transmission power information carried explicitly and/or implicitly by RTS/CTS frames. Before and upon receiving an RTS from the sender, the receiver can measure the interference $P_{I\text{-}current}$ and the power of the received RTS as $P_{rcv\text{-}RTS}$, respectively. Then, it calculates the maximum additional interference it can tolerate, $P_{I\text{-}add}$ by solving

$$SINR = \frac{P_{rcv-RTS}}{P_{I-current} + P_{I-add}} \geq T_{cap} \qquad (2)$$

The receiver puts the calculated $P_{I\text{-}add}$ in CTS frame to advertise it to its neighbors. When a neighbor overhears this CTS frame, it first measures its power. If the perceived power of the CTS is higher than $P_{I\text{-}add}$, this neighbor sets its NAV according to the CTS and stays silent. Otherwise, it ignores the CTS frame presuming that its transmission will not disturb the current reception. Therefore, the parallelism/efficiency is improved by such a "soft blocking" scheme with virtual carrier sensing. Yet at the same time, the collision avoidance is still effective. The method is simple with symmetry assumption. But it only considers one interferer, and the collision may still occur with aggregate interference like in the worst case. This leads to some more sophisticated methods.

In the soft-blocking scheme, the state of a neighboring node is either "on", i.e., in the blocking range, or "off", i.e., out of the range. In contrast to such a simple on-off control, dynamic power control schemes provide more flexible methods for dealing with various interference scenarios in wireless mesh networks. The basic idea can be illustrated as follows. In Figure 2(b), node X is blocked since its transmission with regular power level disturbs the reception at R. However, if node X has a packet for a receiver nearby, say node Y, X may lower its 'voice' (power) so that its interference is below the additional tolerable value for reception at R and yet its power is strong enough for reception on Y. POWMAC considers the additional tolerable interference as a resource, which is shared among multiple concurrent transmissions. Such power and interference information of involved transmissions is exchanged via a series of RTS/CTS handshakes. A POWMAC receiver splits the total tolerable $P_{I\text{-}add}$ across N potential interferers so that the maximum tolerable interference for any single sender P_{MTI} is a fraction of the aggregate interference $P_{I\text{-}add}$. The calculated P_{MTI} is then broadcasted with the CTS frame to neighboring potential transmitters so they can use it to properly set their maximum allowable transmission power.

All the methods introduced in this section for 802.11 MAC highly depend on the accuracy of the propagation model and the interference-error model described before. For implementation, it is imperative for the 802.11 products to measure or control the power with level of accuracy required by these protocols [1].

2.5 Routing metrics in WMN

Originally, the MANET and WMN routing protocols used hop count metric to evaluate paths between sender and destination. It has been noticed that the path with minimum hop may not be the best path because of wide-ranging link performances and the time-variability of wireless links.

There have been some attempts to incorporate various factors related to wireless link quality into the routing metrics. Placing emphasis on the losses and the retransmissions, D. De Couto, et al, [13] proposed Expected Transmission Count Metric (ETX). ETX is the expected or average number of transmissions and retransmissions needed to successfully send a packet in both forwarding and reverse directions between a pair of nodes. The ETX of a route is the aggregate ETX of all the links in the route. Draves et al compared performances of ETX, and other two metrics, per-hop RTT and per-hop packet-pair, against the traditional minimal-hop metric in [15]. The metric of per-hop RTT considers the round trip time (RTT) of packets over a path, and the metric of per-hop packet-pair measures the arrival interval of a pair of packets traversing along a path. Both of the metrics are considered carrying some information of path quality. The experiments [15] show that those link-quality-based metrics, especially ETX, outperform the minimum-hop strategy when all nodes are stationary, while in mobile settings, the minimum hop metric performs better. The reason is that the link quality metrics cannot be updated quickly enough to the changes in highly mobile environment.

Other than loss ratio, some important factors are missing in ETX for evaluating the link performance. The link bandwidth is important for route selection since it could range widely in multi-rate ad-hoc networks. Inverse Rate Weight and Medium Time metric [5] assigns each link a weight which is inversely proportional to the transmission rate of the link, i.e., proportional to the packet transmission time. However, they only consider the raw transmission rate, instead of link effective bandwidth that takes into account the packet losses. In [16], the link bandwidth and rate ranges are incorporated in the metric called Weighted Cumulative Expected Transmission Time (WCETT). WCETT even introduces another factor, channel diversity, dedicated to multi-radio networks.

It is also suggested to consider local medium load when calculating the path metric. The medium load describes the proportion of time when the local channel is busy. However, medium load is a highly dynamic variable and can be affected inversely by the routing decisions. Such coupling makes the routing behaviors extremely difficult to control. An example is the route oscillation problem, which was identified long ago in load-sensitive routing for Internet.

The problem of what factors should be included in the path performance calculation in WMNs is still under study. Nevertheless, it is acknowledged that routing with link-quality-based metrics is effective and practical for establishing high-throughput paths in WMNs. In Chapter 5, we also provide a method to evaluate actual 802.11 link bandwidth, which is also a good metric candidate for routing in WMNs.

2.6 Wireless channel/spatial-diversity

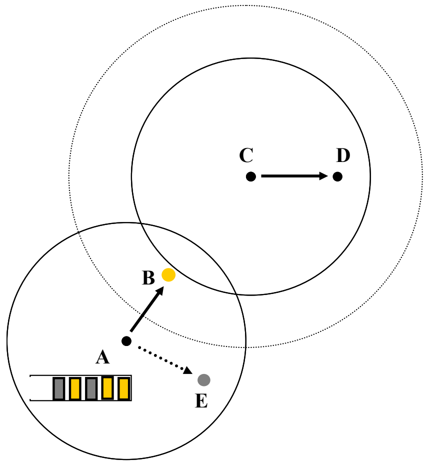


Figure 3 Rescheduling for Head-of-line blocking problem

Another type of methods [27][52][58][59] for increasing concurrent transmissions and improving parallelism in mesh networks is to exploit the channel/spatial-diversity by re-scheduling the frames in the sender's queue. In WMNs, some stations can be particularly overloaded. For example, a mesh gateway needs to deliver simultaneously multiple down-stream data flows between the Internet and many wireless stations; a mesh router may have to serve several neighbors by forwarding their packets along multi-hop paths. The efficiency of such stations is critical to the capacity of a mesh network. However, the performance of the regular 802.11 MAC protocol is susceptible to the head-of-line (HOL) blocking problem.

The HOL blocking problem occurs when the frame currently at the head of the queue in the sender's MAC layer cannot be transmitted successfully due to, say, the temporary unavailability of the receiver. This frame has thus been blocking the subsequent frames from being transmitted although their receivers may be available at this time. For a loaded mesh router or gateway, HOL blocking problem could result in serious throughput degradation.

A straightforward solution for HOL problem is to reschedule the frames in sender's queue based on the status of their next-hop stations. For example, in Figure 3, node *B* is unavailable for receiving any frames from *A* since it is blocked by another transmission. Instead of waiting for *B*, node *A* may first send the frames queued for other available receivers, such as *E*. As a result, the backoff overhead is avoided and the channel utilization is improved. Moreover, as the example shows, the number of concurrent transmissions is increased.

A multicast RTS/CTS (MRTS) handshake is proposed in [27] to obtain the state information of the receivers. An MRTS, in contrast to a unicast RTS in conventional RTS/CTS, is directed to a list of receivers. That is, an MRTS frame contains a list of next-hop receivers for which the sender has DATA packets currently queued. Each element of the list contains a receiver's address and the NAV of its corresponding packet. The priority among different receivers is decided by the order in which the receivers are arranged in the MRTS frame. That is, the earlier a receiver's address appears on the MRTS list, the sooner this receiver can return a CTS. Only the first available receiver on the list can return a CTS to the sender. Then, the sender retrieves the corresponding frame from its queue and transmits it to that receiver. Since the MRTS probes the availability of multiple receivers almost simultaneously, the likelihood of MRTS failure, i.e., no receiver available, is low. Hence, the idle time due to backoff on the loaded stations can be significantly lowered and their utilization is improved.

Similar re-scheduling scheme is introduced in [38] to mitigate inter-flow interference among the traffics originated from a mesh gateway. The distance/received signal strength between each pair of nodes are collected and reported to the mesh gateway by signaling. Then, the distance between each pair of down-stream flows are calculated and a virtual coordinate is constructed. Based on the coordinate information, the mesh gateway evaluates the mutual interference among the paths between itself and the other end stations. Then, the mesh gateway schedules back-to-back the packets which will take the paths with minimal mutual interference.

The methods above exploit the temporal-spatial diversity in the network by rescheduling packets on a loaded sender. Such diversity can particularly be found in a dense and busy mesh network.

2.7 Wireless network coding

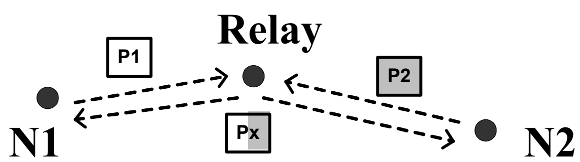


Figure 4 An example of coding

Wireless network coding is a fundamentally different approach from other attempts to improve the throughput of wireless networks. In contrast to other approaches, which utilize the medium better when it is not fully congested, but do not increase its capacity, network coding is able to increase network capacity [3]. It changes the way that information has been treated for a long time in that it can be spread and combined when transported [3].

All of the work on network coding is primarily theoretical and assumes multicast traffic until COPE [30] presents a practical implementation to achieve higher unicast throughput in multi-hop wireless networks. The basic idea of COPE can be illustrated by the example in Figure 4.

Suppose that node *N1* has a packet *P1* for node *N2*, and that node *N2* has a packet *P2* for node *N1*. Assume that both of these packets must be relayed by a replay. Without network coding, four transmissions will be needed to achieve such a message exchange. By following procedure, however, it can be done with only three transmissions. After receiving *P1* and *P2*, the replay broadcasts a packet *Px* obtained by XOR-ing the original two packets. Upon receiving *Px*, receivers *N1* and *N2* can obtain each other's packet by XOR-ing *Px* again with their own packet. In this way, a single transmission can benefit multiple receivers and the coding thus increases the network capacity. The advantage of coding here is accomplished through the broadcasting nature of wireless links.

COPE also describes the conditions of some other more complicated coding scenarios, like cross topology, X-topology and wheel topology. To find the coding opportunities in these scenarios, COPE makes nodes snoop on all communications and store all the overheard packets for a short period. They exchange the information of what packets they have with each other. Then, each forwarder makes decisions of what packets should be combined, making sure the receivers have corresponding packets to successfully decode the transmission.

The coding gain of the scenario in the above example is 4/3. That is, with coding, three transmissions achieve the same exchange that requires four transmissions without coding. The actual throughput gain could be higher. In the example scenario, the relay node is the bottleneck and the packets transmitted by *N1* and *N2* are backed up at the relay node's queue. Coding allows the relay to drain the queued packets twice as fast, therefore doubling the throughput. This is referred to as MAC gain.

We further introduce the background of network coding and the related work in Chapter 4.

2.8 Opportunistic routing

Like wireless network coding, opportunistic routing is another inspiring attempt made recently to achieve high unicast throughput in wireless multi-hop networks utilizing the broadcasting nature. The first opportunistic rotuing protocol, ExOR, is proposed in [8], and extended lately in [10].

In ExOR, similar to anycasting, any neighbor en route can forward an overheard data packet as long as it is determined that such an opportunistic forwarding gets the packet closer to its destination. This method is fundamentally different from the traditional routing protocols, which find and use the fixed path between source and destination. Such innovation enables a packet to travel by long hops incidentally when the channel condition is good, and ensures the transmission using closer neighbors otherwise. In addition, ExOR takes advantage of the packet redundancy in the network to achieve robustness. That is, when a forwarder en route failed to receive a packet, one of other forwarder candidates, which overheard the packet when it was transmitted by the previous hop, can take over the forwarding task. This saves the bandwidth waste on retransmissions, especially under lousy channel conditions.

With multiple forwarder candidates, it is likely that a packet be forwarded more than once for each hop. Thus, a mechanism to avoid duplications is needed. In ExOR, each node maintains a routing table with ETX as metric via periodic link-state flooding. The source specifies a forwarder list in priority order based on the ETX from each potential forwarder in the list to the destination. A node is included in the list only if the node is closer to the destination than the source. After the source has sent a batch of packets, the participating nodes send their overheard fragments in the order in which they appear in the list. By such scheduling, ExOR makes sure that only one node transmits at each time. A node starts sending at the time it predicts the previous fragment will finish. The closer a node to the destination, the less time it waits for. A forwarder will not repeat those fragments it overheard from other higher-priority nodes and only send the fragments missed by them. In this way, ExOR makes packets travel by long hops opportunistically and at the same time limits the duplications.

ExOR outperforms the traditional routing methods significantly because it effectively exploits the wireless broadcasting nature and the consequential packet redundancy in the network. By our proposed method in Chapter 4, we show that above features can both be used to increase network coding opportunities in the network.

2.9 Other work

Some other related work on improving mesh network performances include MAC-layer rate adaption [22][55], TCP wireless congestion control and so on. Recently, there has been a great amount of research on multi-radio multi-channel mesh networks. Corresponding protocols have been designed to assign multiple non-overlapping channels dynamically for the purpose of parallelism when nodes are equipped with multiple radio devices.

In this chapter, we show various approaches on improving the throughput in WMNs. In the rest of the dissertation, we present our efforts toward high throughput gain in 802.11 WMNs.

Chapter 3

Channel State Diversity and Channel-State-Based Scheduling

In section 2.6, we introduced MRTS, a promising solution for HOL blocking problem of 802.11-based WMNs. To maximize the benefit of MRTS, we propose a channel-state-based scheduling scheme that adapts both the content and the length of the receiver list in a MRTS frame to dynamic network conditions. Our strategy constructs an MRTS list of receivers that have mutually diverse channel states. This leads to high acceptance ratio of MRTSs with reduced overheard of MRTS frames.

The rest of the chapter is organized as follows. Section 3.1 provides an overview of the MRTS mechanism and its overheard. Section 3.2 presents our basic idea and design. Simulation results are given in Section 3.3 to demonstrate the benefits of the proposed protocol. Section 3.4 concludes the work.

3.1 Background

In 802.11, each time a DATA frame or Request-to-Send (RTS) transmission times out, the sender doubles the contention window, to wait for a longer backoff time before retransmission, for the purpose of collision avoidance. This DATA frame will not leave the queue until the transmission is acknowledged or until the maximal number of retries is reached. This frame has thus been blocking the subsequent frames from being transmitted although their receivers may be available at this time. This is the so-called head of line blocking (HOL) in 802.11 MAC. Due to the exponentially-growing backoff time overhead, HOL blocking can lower greatly channel utilization and network capacity. Our simulation indicates that the fraction of backoff time at the sender's MAC layer may reach up to 70%. For a loaded mesh router or gateway, HOL blocking could result in a serious congestion problem. During the backoff process of a mesh gateway, more and more frames could arrive from a wired Internet connection and be blocked in the queue. With more frames arriving and the head frame blocking the queue, the gateway eventually gets overwhelmed and the queue overflows and starts dropping packets. This may further trigger an upper layer (e.g., TCP) backoff, leading to further throughput degradation. Thus, in order to improve the performance of multi-hop mesh networks, the HOL blocking problem must be addressed.

3.1.1 Overview of MRTS

Most attempts at addressing the HOL blocking problem are based on basic access scheme or unicast RTS [6][18]. Conversely, an innovative solution is to extend it to a multicast case called MRTS [27][52]. That is, the sender includes in RTS frame the addresses of multiple neighbors for whom it has data frames ready in the queue, and probes them by this multicast RTS.

The IEEE 802.11 MAC layer specifies a CSMA/CA-based protocol, enhanced with an RTS/CTS/DATA/ACK handshake for virtual carrier sensing. The RTS/CTS dialogue is used to reserve channel on both the sending and receiving sides. Originally, the RTS frame is addressed to a single receiver. An MRTS, in contrast, is directed to a list of receivers. That is, an MRTS frame contains a list of next-hop receivers for which the sender has DATA packets currently queued. Each element of the list contains the receiver's address and the NAV of the corresponding packet. The priority among different receivers is decided by the order in which the receivers are arranged in the MRTS frame. That is, the earlier a receiver's address appears on the MRTS list, the sooner this receiver can return a CTS. The top candidate receiver that successfully receives MRTS replies with a CTS, unless it is blocked by an ongoing transmission in its neighborhood. If a lower-priority candidate detects that all higher-priority candidates remained silent for a defined period of time, it has the right to reply with a CTS (Figure 5). Such a right-to-reply is implicitly propagated down the chain until a non-blocked receiver sends a CTS or all receivers remain silent and the sender times out. The sender

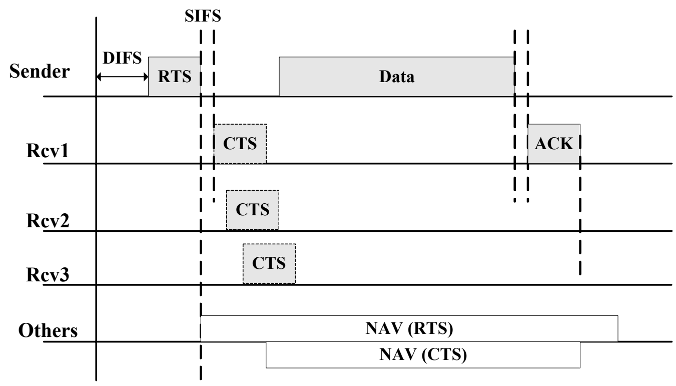


Figure 5 MRTS protocol timeline

finds the responding receiver's address from the CTS reply. Then, the sender retrieves the corresponding packet from its queue and transmits it to that receiver. The dialog ends with an ACK from the receiver if the transmission is successful.

An MRTS is successfully accepted as long as one receiver in the list replies. Including more receivers in the MRTS list helps increase the probability of acceptance and thus the utilization of the sender. Consider the case where an MRTS fails, i.e., all of the receivers in its list remain silent. The likelihood of this is lower for longer lists of receivers. Moreover, with MRTS, multiple senders can select the receivers from the lists for next transmissions so that they can coexist. This increases parallelism in the network.

However, there is a cost associated to the long MRTS list. The longer control frame causes higher transmission overhead and increases the likelihood of collision.

3.1.2 Overhead of MRTS

For each receiver in the list, 8 bytes are added to the RTS frame (6 for the address and 2 for the NAV duration). For example, a 4-node MRTS list adds 24 bytes (or 192 bits), i.e., an additional 192μs to the RTS transmission time in 802.11b, where RTS/CTS is transmitted at the basic rate of 1Mbps. This overhead is not trivial especially when the DATA frame is transmitted with higher-rate modulation, e.g., for 500-byte DATA frames at a rate of 11Mbps, the total duration of an RTS/CTS/DATA/ACK cycle with a 4-node list MRTS is 13.8% longer than with a unicast RTS. In addition to the overhead of transmission time, a longer MRTS frame has higher chance of collision. To avoid collisions, an exponential backoff mechanism is used in 802.11. A station ready to transmit RTS has to wait for a DIFS and a random amount of time between 0 and the contention window time, T_{CW} = (CW) × SlotTime. CW is set to 31 for the first attempt and is approximately doubled for each unsuccessful subsequent attempt. Hence, if two hidden stations attempt to access medium, the probability of no RTS collision can be approximated by

$$\begin{aligned} P &= \Pr\left(\left|X - Y\right| > T_{RTS}\right) \\ &= 2 \cdot \int_{y=T_{RTS}}^{T_{CW}} \int_{x=0}^{y-T_{RTS}} \frac{1}{T_{CW}^{2}} dxdy = \left(1 - \frac{T_{RTS}}{T_{CW}}\right)^2 \end{aligned} \quad (3)$$

where X and Y are the random variables between [0, T_{CW}] picked by two stations respectively and T_{RTS} is the transmission time of RTS. For the first attempt, the probabilities of no collision using unicast RTS vs. a 4-node MRTS are 0.187 and 0.015, respectively. These probabilities drop sharply for longer MRTS lists.

The extra addresses contained in the MRTS add to the control overhead, particularly if a gateway serves a large number of mobile nodes. Thus, efforts are imperative to reduce such overhead. On the other hand, over-limiting the MRTS list length may hinder the capability of multiple-receiver probing, thus hindering MRTS' effectiveness.

Thus, a more careful construction of the MRTS receiver list is needed to reduce the overhead while maintaining its effectiveness simultaneously.

3.2 Adaptive Channel-State-Based Scheduling for MRTS

3.2.1 Basic idea

It is noticed that the idea of MRTS exploits the diversity of receivers' channel states in the list. We observe that geographically-proximal stations are likely to share similar channel states. Adding those similar stations to the list does not necessarily increase the diversity. The term "channel state" is referring to MAC layer condition rather than physical channel condition. This means that a receiver's channel state is "good" if it is idle (no concurrent transmissions). If high correlation of channel states is observed for two candidate receivers, this implies low diversity between them, and thus it is unnecessary to include both of them in the MRTS list. For example, suppose that node *O* in Figure 6 is delivering packets in four flows through its neighbors. Receivers *A* and *B* are both in the carrier-sensing range of station *X*, i.e., their channel states are synchronized to *X*'s behavior. When *X* is transmitting, both *A* and *B* are in "bad" state and are unable to reply *O*'s request. For sender *O*, the probability that *A* and *B* are both in good channel state is the same as for one of them to be in a good state; likewise for nodes *C* and *D*. Thus, we can achieve the same level of effectiveness as the MRTS which includes all nodes by using a shorter node list and, thus, smaller overhead, by selecting the nodes with diverse channel state patterns in the list, e.g., $\{A, C\}$ or $\{B, D\}$.

From above observation, we speculate that, if the receivers are chosen not

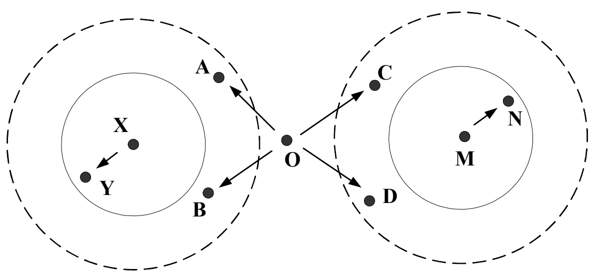


Figure 6 An MRTS example

randomly but based on the knowledge of their channel-state correlations, so that they are likely to have diverse channel states, then a short list can achieve the same effectiveness in HOL blocking avoidance as longer lists.

Indeed, the contribution of our study is two-fold. On one hand, we show that the diversity of receivers can be collected and quantified by the sender without additional overhead. Second, with such information of channel states of its neighboring receivers, intelligent decisions can be made to fulfill various purposes, including efficient scheduling presented here, localizationing/positioning, routing and so on.

We first present our idea about how the channel diversity information can be obtained. Second, we show how to select a subset of neighbors for the MRTS using the above recorded information. Furthermore, we enable each node to make its own decision on how long this list should be, depending on the current network condition.

3.2.2 Channel diversity estimation

In this section, we show that the correlation/diversity of channel states between a pair of receivers can be obtained merely by observing the historical outcomes of MRTS requests.

We assume that a node can include up to L neighbors in its MRTS. The outcome of an MRTS of length L is that only the r-th neighbor replies with a CTS, where $1 \le r \le L$. That is, L neighbors of the sender were polled in the order specified by the list contained in the MRTS. The sender can tell that none of the first $r-1$ neighbors in the list was able to reply, i.e., all are in bad channel state, while the r-th neighbor is in a good state and is able to reply with a CTS. We use the case $r = L + 1$ as notation for the special case when no node in the MRTS list replies. We call such a value r the *rank* of the MRTS. For an MRTS of rank r, the relevant information is that the r-th neighbor is in a different (i.e., better) channel condition than every node i ($1 \le i < r$) and that nodes i and j ($i \ne j$, $1 \le i < r$ and $1 \le j < r$) are in the same bad channel state.

The sender maintains a table with two counters S_{ij} and N_{ij}, for each pair of its neighbors i and j, to record the number of occurrences of the above difference among the outcomes. S_{ij} denotes the numbers of occurrences in historical records where i and j are receivers included in an MRTS and i appears before j in the list, but only j, the latter, was able to reply. S_{ij} indicates how diverse i and j's channel states are. Similarly, N_{ij} counts occurrences when both i and j are included in an MRTS frame but neither was able to reply, which means that they are simultaneously in a bad state for N_{ij} times. These counters are updated every time a new observation is made. When the total number of observations grows large, a new observation makes insignificant difference in the estimated weight. Therefore, a sliding window is used to increase the agility of adaptation. The window keeps only M most recent observations for each pair of receivers, where M is the size of the window. The counters record only the observations in the window. The size of the window can be adjusted to match the factors affecting the channel state, such as, average session lifetime and the movement pattern of the stations. The sliding window size is set to 20 in our simulations.

To maintain the diversity counters so to reflect the channel diversity among the neighbors of a transmitting node, this node updates the entries of the table according to the observation that it has made based on the rank r of the latest MRTS frame ($1 \le r \le L$).

Specifically, we increase the counter S_{ir} by one for every i $(1 \leq i < r)$. In addition, we increase the counter N_{ij} by one for every i and j ($1 \leq i < j < r$).

3.2.3 Adaptive Scheduling

To utilize the above history information, we define a value, called *diversity weight* W_{ij}, for each pair of a sender's neighbors i and j to represent how uncorrelated or diverse receivers i and j are in their channel states:

$$W_{ij} = W_{ji} = \frac{S_{ij} + S_{ji} + 1}{S_{ij} + N_{ij} + S_{ji} + N_{ji} + 1} \tag{4}$$

The sum of counters S_{ij} and S_{ji} represents how many times i's and j's historical channel states are different. The denominator is the size of the whole sample space including the counters of instances, N_{ij} and N_{ji}, when both receivers' states may be influenced and synchronized by the same or similar traffic pattern. Thus, W_{ij} indicates normalized state diversity between i and j. We add one to both the numerator and denominator for initialization when the counters are zero.

When a node has a packet to send, it constructs an MRTS of length L. The neighbors to be included in the MRTS list are selected as follows. It first selects a neighbor, uniformly at random, denoted by N_1, for which it has packets in the queue. To include the second neighbor, it calculates the diversity weight between node N_1 and every other neighbor for which it has packets queued. It then selects a node among these neighbors with likelihood proportional to the diversity weights calculated. That is, the neighbor with a packet available and with highest diversity weight relative to node N_1 has the highest probability to be selected. Denote it N_2. To include the third neighbor, it calculates the *combined diversity weight* for each neighbor j for which it has packets available, relative to nodes N_1 and N_2. The combined diversity weight is defined as $W_{jN_1} \times W_{jN_2}$. Again, it then selects a node among these neighbors with likelihood proportional to this combined weight. The likelihood of a neighbor appearing early on in the list is proportional to its contribution to the diversity measure. We denote such a selected node N_3. Generally, to include the i-th neighbor into the MRTS ($2 < i \leq L$), it calculates the combined diversity weight for each node j for which it has packets available, relative to the $i-1$ neighbors already included in the list. This weight is defined as $\prod_{m=1}^{i-1} w_{jN_m}$. It then selects the i-th neighbor to be included in the MRTS randomly with likelihood proportional to the weight. Thus, the node completes the construction of the list of L neighbors to be included in the MRTS.

Note that the diversity weighting described above attempts to maximize the diversity of channel states of the receivers included in the list. In addition, the randomization ensures that no combination of receivers is completely excluded even if a combination is highly correlated at the moment. This is important for future updates when channel states are changed. Further, such randomization promotes fairness by avoiding starvation of any flow.

3.2.4 Determination of List Length

In order for the nodes to adapt to the channel conditions, we further enable each node to decide on how many nodes to include in the MRTS depending on the updated network conditions. To do that, each node maintains a running average of the ranks of the MRTS frames that it has attempted, denoted $\bar{r}$. When the rank r of a new MRTS is recorded, $\bar{r}$ is updated with $0.875\bar{r} + 0.125r$. Using this average rank, the MRTS frame that a node constructs always has a length $l = round(\bar{r}) + 1$. Such length adjustment can be seen as a feedback control mechanism. That is, when the average rank of the MRTS outcomes becomes large as the network condition changes, the list will be extended to maintain a high level of diversity. The small constant (i.e., 1 in the above definition) added to the average rank enables the adaptive list growth. If the network condition changes in the opposite direction so that some receivers in the current list do not contribute to the diversity value, those receivers will be scheduled later than the others since our channel-state-based scheduling puts ahead the receivers with the highest combined diversity weight. Therefore, the average rank will decrease. By observing such decreases, the mechanism shortens the MRTS list automatically.

3.3 Simulation results

We test and compare our fixed-length and variable-length channel-state-based schemes against the random scheduling schemes in the basic MRTS using ns-2 with default PHY layer settings. Three scenarios are designed below for this purpose. We show in Scenario 1 that the channel-state-based (CSB) scheme with a short fixed-length MRTS receiver list can outperform the random scheme with an equal or even longer list. By scenario 2, we test the adaptability of our channel-state-based scheme to mobility. In Scenario 3, we show that an appropriate list length is adjusted adaptively to the dynamic traffic situations in the network. Our variable-length channel-state-based (VL-CSB) scheme achieves the highest throughput, compared to the fixed-length random scheme and the "include-all" scheme.

3.3.1 Scenario 1

We first test our fixed-length channel-state-based scheme with the random scheduling schemes using ns-2 on the example scenario in Figure 6. The link rate is set to 11Mbps. The rates of CBR flows *OA*, *OB*, *OC* and *OD* are all set at 650 pkts/sec so that node *O* always has packets in the queue for each flow. We set the rate of flow *MN* to 0 at first. Then, we vary the sending rates of flow *XY* from 100 to 500 pkts/sec. For each rate, we perform 50-second simulations with 2-node list random scheme, 4-node list random scheme and the 2-node channel-state-based scheme, respectively. In Figure 7(a), the channel-state-based scheme outperforms both the 2-node and 4-node random schemes for all rates of flow *XY*. The aggregate throughput of the 2-node random scheme declines as the rate of flow *XY* grows, while the ones of channel-state-based scheme and 4-node random scheme remain constant. In this scenario, the receivers of *O* are grouped into two subsets $\{A, B\}$ and $\{C, D\}$. Nodes *A* and *B* are within the carrier-sensing range of node *X*, and they have the same channel state most of the time. For 2-node random scheme, there is a 1/3 chance that two nodes from the same subset are selected, which leads to low channel-state diversity in MRTS frames. In contrast, the channel-state-based scheme and the 4-node scheme achieve higher MRTS success rate and thus less backoff overhead by maintaining the high channel-state diversity in the receiver list. Figure 7(b) shows that the backoff time fractions on *O* for channel-state-based scheme and 4-node scheme are much lower than for the 2-node random scheme, especially when the rate of *XY* flow is high.

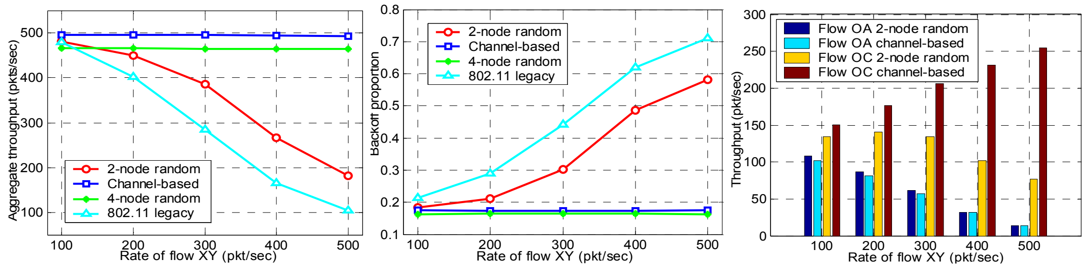


Figure 7. (a) aggregated throughput; (b) backoff time proportion; (c) Individual throughputs; Rate of MN is 0

Both channel-state-based scheme and the 4-node random scheme achieve the same level of diversity because the success rate of an MRTS with two nodes, each selected from different subsets, is the same as that of a 4-node MRTS. The constant gap between the channel-state-based and 4-node schemes in Figure 7(a) and (b) can be attributed to the extra overhead in transmission time of 4-node MRTS's. Figure 7(c) shows the throughputs of individual flows *OA* and *OC* with the 2-node random scheme and the channel-state-based scheme. With channel-state-based scheme, the throughput of flow *OA* is slightly lower but the throughput of flow *OC* is greatly improved. This confirms that avoiding selecting both *A* and *B* in MRTS list reduces backoff- and retransmission overhead and alleviates the HOL-blocking problem, especially when the rate of flow *XY* is high. Figure 7 indicates that the local network capacity can be significantly increased by an appropriate construction of the MRTS list.

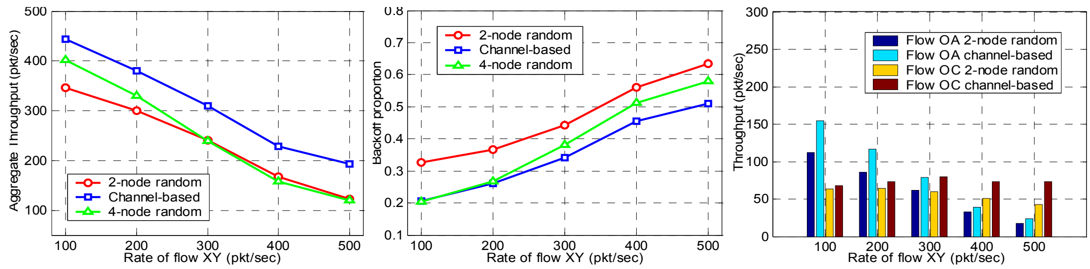


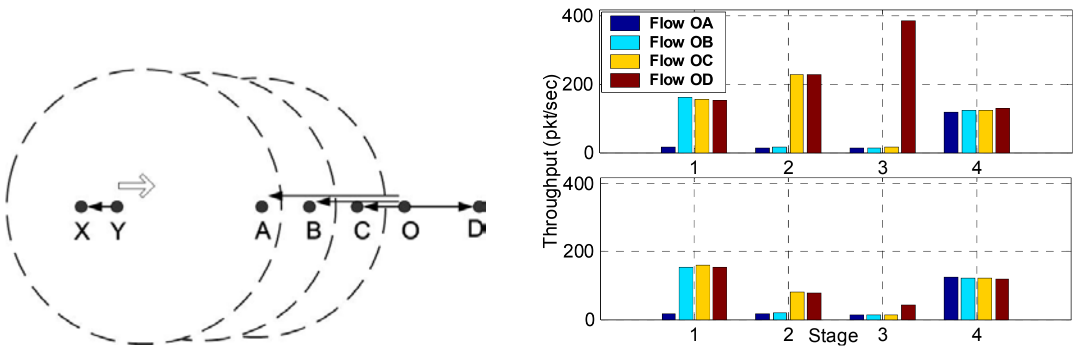
Figure 8. (a) aggregated throughput; (b) backoff time proportion; (c) Individual throughputs; Rate of MN is 300

We then set the rate of another interfering flow *MN* at 300 pkts/sec and repeat the previous tests. The results are shown in Figure 8. The network capacity now is not enough to sustain all 6 flows and the aggregate throughput on node *O* slips for all schemes when rate of flow *XY* increases. The aggregate throughput of node *O* with channel-state-based scheme is consistently higher than for the other two schemes. In high load cases, the throughput of 4-node scheme drops drastically as the overhead of high collision rate due to longer MRTS frames becomes dominant. Also, notice that the throughputs of all individual flows in Figure 8(c) are increased with the channel-state-based scheme. It means that, in such overloaded scenario, every flow benefits from the backoff overhead cut achieved by the channel-state-based scheme.

3.3.2 Scenario 2

In wireless networks, the local channel conditions can be changed by movement of nodes. In Figure 9(a), as node *Y* moves toward *O*, nodes *A*, *B* and *C* fall into *Y*'s interference zone, in order. In this scenario, we test the adaptability of our channel-state-based scheme.

The results are shown in Figure 9(b). In stage 1, where only node *A* is under the interference of node *Y*, the throughput of *OA* is low and receivers *B*, *C* and *D* share most



(a) (b)

Figure 9. (a) scenario 2; (b) channel state based scheme vs. 2-node random

of the local bandwidth. In stage 2, where *A* and *B* are in the interference zone, the HOL-blocking problem lowers the throughput of flows *OC* and *OD* for the 2-node random scheme. The channel-state-based scheme detects the increasing correlation of *A*'s and *B*'s states, and the probability of scheduling them in the same MRTS is reduced accordingly. Thus, flows *OC* and *OD* obtain greater fraction of serving time on *O*. Likewise, in stage 3, where *A*, *B* and *C* are under interference, frames of flow *OD* are more frequently scheduled by the channel-state-based scheme. This again alleviates the HOL-blocking problem. The throughput of flow *OD* (upper graph) is 3 times higher in stage 2 and 7 times higher in stage 3 than for the random scheme (lower graph). Lastly, as we stop flow *YX* in stage 4, both schemes assign the resource equally for all four receivers (*A*, *B*, *C* and *D*). The results show that the channel-state-based scheme is responsive to channel state changes and more efficient than the random scheme.

3.3.3 Scenario 3

In this scenario (Figure 10), we add 4 neighbor nodes of *O* and two more interfering flows. We start the 4 interfering flows at different times of the simulation,

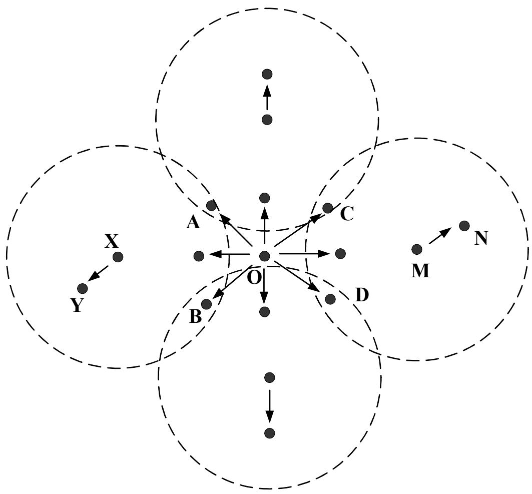


Figure 10 Scenario 3

emulating a more realistic dynamic network environment. The rate of each interfering flow is fixed at 400 pkts/sec. The flow on the left is started first, followed by the other three in the counterclockwise order, one in each stage of 50 seconds. So, for stage *n*, there are *n* interfering flows running. In such a complex and dynamic scenario, fixing the length of MRTS list for different situations permanently is apparently not a good solution. When few interfering flows are present, a long list may cause redundancy and lead to high overhead as the results of Scenario 1 show. Conversely, a short list is insufficient to leverage the channel state diversity of the list as more interfering flows are started.

The experimental results (Figure 11) show that, by combining the list-length adaptation with channel-state-based scheduling, our VL-CSB scheme achieves the best throughput performance in all stages, compared to the fixed-length (4-node) random scheme and 8-node "include-all" scheme. The latter always includes all the active next-hop receivers of the sender and it is taken in the comparison to illustrate the inappropriateness of using excessively long MRTS lists.

Figure 11(a) shows how the average rank, observed by the sender at the network center, grows as the number of interfering flows increases at every stage in the VL-CSB result. Every time an additional interfering flow starts, the acceptance rate of the MRTS drops and the rank increases. Accordingly, the VL-CSB grows the receiver list to maximize the MRTS acceptance probability. Without such a length-adjustment mechanism, in Figure 11(b), the 8-node "include-all" scheme's performance suffers in the first two stages when there is a low diversity of channel states in the neighborhood. This is due to the overhead of transmitting its excessively long MRTS frames. In the last two stages, its advantage of high diversity owing to the long list overcomes the transmission overhead, and its performance approaches VL-CSB's. These results show that the length-adjustment mechanism effectively lowers the overhead of MRTS. In addition, the VL-CSB outperforms the 4-node random scheme significantly in stages 2 and 3. This means that the random scheduling cannot achieve the same degree of channel-state diversity in the list as our channel-state-based scheduling or the "include-all" scheme. The 4-node random scheme lowers the MRTS success ratio. In contrast, the CSB scheme makes it possible to use shorter lists to achieve higher diversity. According

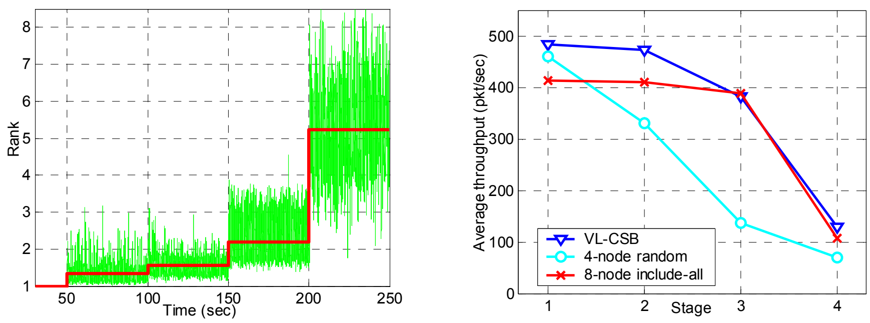


Figure 11 (a) Average rank; (b) aggregate throughput to Figure 11(a), the length of MRTS chosen by VL-CSB is less than 4, i.e., mostly 2 or 3 in stages 2 and 3. The performance boost by VL-CSB against the 4-node random scheme, see Figure 11 (b), suggests higher diversity in the lists constructed by CSB.

3.4 Conclusion

In original MRTS scheme, the diversity is achieved by keeping a relatively long receiver list. It can be speculated that when more neighbors are included in the MRTS, the likelihood that all of them are busy is lower. However, the margin of such a higher success rate will decrease and even become negative when factoring in the overhead of having a long MAC header with a large number of neighbors included. In this chapter, we present a variable-length, but short, list in the MRTS to achieve high network performance in general settings.

Earlier work on MRTS includes a subset of a transmitting node's neighbors in the address list of the frame in order to increase the transmission success rate of the RTS. The determination of such a subset, however, is somewhat arbitrary. We show that the effectiveness of the MRTS could be significantly improved if nodes make the decision with more judgment. Indeed, with the information of channel states and statuses of its neighboring receivers, more intelligent decisions can be made by the transmitting node. The multicast characteristic of MRTS measures the channel conditions of multiple receivers simultaneously. Based on the observed responses of MRTS's, the sender can estimate the neighbors' channel states and their correlations. Furthermore, we use such information not only to select the receivers but also to adjust the length of the receiver list in the MRTS to adapt to the network conditions. The effectiveness of introducing such a notion is supported by a set of experiments, which indicate that an intelligent inclusion of the receivers provides higher throughput than randomly including part of or all of them.

Our solution, dedicated to single-channel environments, can co-operate with other multi-radio/multi-channel MAC protocols to further enhance the capacity of wireless mesh networks. The MRTS with its extension can mitigate the blocking time while operating on each individual channel. Moreover, as shown above, the spatial and channel proximity of receivers can be estimated by the sender through their historical responses to MRTS. Such information may be helpful for improving efficiency of a channel assignment scheme by assigning non-overlapping channels to the proximal stations to reduce mutual interference. This feature of MRTS and a combination with multi-channel mesh networks represent our future work

Chapter 4

Opportunistic Forwarding for Network Coding

Network coding is an innovative technique for improving potential network throughput and robustness. It changes the way that packets have been treated for a long time in that they can be combined when transported and separated when received [3]. One important application of network coding is data communication, i.e., to achieve high network capacity. Most of the previous work studies its advantage for multicast or broadcast traffics. COPE [30] presented the first practical implementation of network coding to improve the efficiency of unicasting in multi-hop wireless networks. The experiments [30] show that COPE can provide a several-fold increase in the throughput of wireless mesh networks. However, with traditional fixed single-path routing, the mixing/encoding can only be achieved at the joint points of two end-to-end flows. This limits the coding opportunities in the network. We present a new protocol called BEND, which can proactively capture coding opportunities in 802.11-based mesh networks. It achieves high coding transmission ratio by exploiting the wireless broadcasting feature and the consequent redundancy of packets in the local area of a network. Our simulation results indicate significant throughput gain of BEND.

4.1 Background

4.1.1 Network coding in wireless networks

Network coding is a relatively new research area in communication networks. It enables data flows to approach the Shannon Capacity Limit individually by splitting and

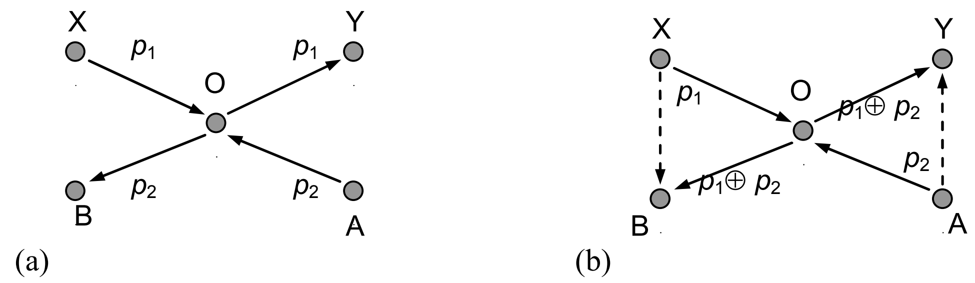


Figure 13 Wireless network coding illustrated: (a) regular exchange; (b) coded exchange

combining information at intermediate nodes in the network. Such operations on information flows can be implemented as simple linear combinations over some finite field. Two fundamental benefits of network coding are greater throughput and higher robustness. These in turn translate to bandwidth and energy efficiency and fault tolerance in multi-hop wireless networks. Current research on network coding is transitioning from theoretical frameworks to increasingly practical systems.

The way that network coding increases the throughput of a multi-hop wireless network can be explained using a simple example of a 5-node network (Figure 13) [30]. Here, nodes X, B, and O are within each other's transmission range; so are nodes Y, A, and O. Suppose that node X has a packet p_1 for Y via O and node A has a packet p_2 for B

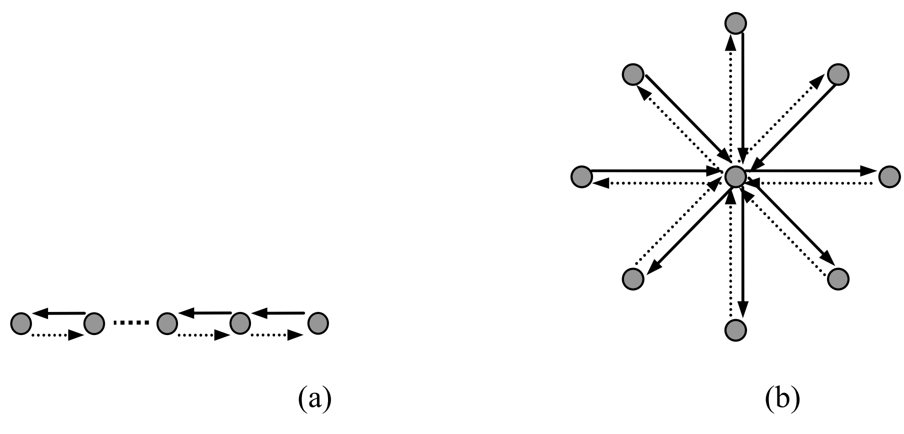


Figure 12 More general coding scenarios: (a) chain (b) wheel

via O. In a traditional non-coding approach (Figure 13(a)), after O's reception of packets p_1 and p_2, it relays these packets separately. Thus, a total of 4 transmissions are required. In contrast, if network coding is used (Figure 13(b)), after O's reception of p_1 and p_2, it transmits XOR combination $p_1 \oplus p_2$ in the wireless channel. Since node B (Y, respectively) is within the transmission range of X (A, resp.), it has also overheard p_1 (p_2, resp.). With node B's knowledge of p_1 (Y's knowledge of p_2, resp.), it can reconstruct p_2 (p_1, resp.) by applying XOR $\oplus$ to the two receptions from X (A, resp.) and O. Consequently, only 3 transmissions are needed for the packet exchange. More generally, network coding can be used in such scenarios as a path transporting two flows in reverse directions (Figure 12(a)) and combining multiple packets (Figure 12(b)).

4.1.2 Coding-aware routing: concentration vs. diffusion

In a previous wireless coding approach, COPE [30], packet mixing can only be performed at the joint nodes of the paths determined by the routing module, such as the focal nodes in Figure 13 and Figure 12. This significantly limits the coding opportunities in the network. Clearly, in order for network coding to be useful in multi-hop wireless networks, there should exist sufficient opportunities to mix traffic flows in the network. Currently, this is achieved by concentrating flows at certain nodes in the network. This could be implemented via centralized code-aware routing [48][44] or using code-aware metrics [54]. As a result, some nodes in the network are favored by the routing module so that much traffic is routed through them for more coding opportunities. However, these approaches can be problematic. First, a network layer implementation of such a "traffic-sensitive" routing protocol is unrealistic in multi-hop wireless networks since traffic flows change over time. Routing that depends on the correlation of dynamic flows has been shown impractical, even in the Internet where the traffic is much more statistically stable over time. Further, concentrated traffic inevitably overloads intermediate nodes in the network. These overloaded forwarders can be a vulnerable point because of higher risk of battery-energy depletion and information leaks. Other problems of traffic concentration include increased queuing delay and thus end-to-end delay, danger of buffer overflow, and further adversary effects to TCP flows. At the link layer, on one hand, traffic concentrated within a neighborhood worsens channel contention in the area. On the other hand, if flows are forced to go through a specific node, this overloaded node is bound to drop packets which it is unable to handle. This is especially problematic in multi-hop wireless networks because dropping packets along a path means invalidating the work performed by earlier forwarders and wasting the network bandwidth already consumed. The benefit of being able to scatter flows through multiple forwarders dynamically at the link layer in a multi-hop wireless network is called "*diffusion gain*" in the rest of the paper.

Indeed, traffic separation rather than concentration has been a key approach to higher throughput in mesh networks. When flows are more evenly distributed in the network, the interference among them is minimized and the network capacity limit can be approached. Hence, traffic concentration in network coding conflicts the need of traffic separation. Does traffic mixing for network coding inevitably imply traffic concentration? Not really.

4.1.3 MAC-layer proactive mixing

In this chapter, we present BEND [61][62], a MAC layer solution to practical network coding in multi-hop wireless networks. It is the first exploration of the broadcasting nature of wireless channels to proactively capture more coding opportunities. As a matter of fact, the result of a node's transmitting a packet is that all of its neighbors can potentially receive it, and such redundancy of packets should and can be utilized. In BEND, any node in the network can code and forward a packet even when this node is not the intended receiver of the packet, if the node believes that doing so it can lead the packet to its ultimate destination. Essentially, BEND considers the union of the contents of the interface queues of the nodes within a neighborhood collectively, i.e. a "neighborhood coding repository", whereas traditional mixing methods, e.g. COPE, only process "individual coding repositories" at separate nodes. Our experimental evaluation shows that BEND creates significantly more coding opportunities in a dynamic and adaptive fashion with minimum assumptions on the routing protocol compared to prior work. The contributions of BEND are:

(a) It makes network coding practical by proactively seizing such opportunities and by using them intelligently.

(b) This is achieved without concentrating traffic flows or overloading specific nodes. It exploits the broadcasting nature of wireless channels by utilizing redundant packet copies within the proximity of a node. In this way, it achieves both diffusion gain and coding gain, which are conflicting in the existing solutions.

(c) It exploits another dimension of multi-user diversity in wireless networks. Multi-user diversity has proved to be effective in achieving higher aggregate system performance in wireless communications. Here in BEND, multi-user diversity is in the sense of diversity of queue contents at different forwarders.

The rest of the chapter is organized as follows. In Section 4.2, we review the basic idea of BEND to help readers with the subsequent relatively involved details. We then highlight the design objectives of BEND and the challenges in Section 4.3.1. The design details are presented in Section 4.3.2. The effectiveness of BEND is tested by the experiments in Section 4.4. After digesting the details, the readers are walked through a discussion in the context of some recent related work on practical network coding and on exploration of the broadcasting nature in multi-hop wireless networks in Section 4.5. We conclude this paper in Section 4.6 and speculate on future research to further explore BEND.

4.2 Basic idea

The gist of BEND is to utilize overheard packets that are otherwise discarded in conventional networking protocols. In a network supporting multiple flows, there are

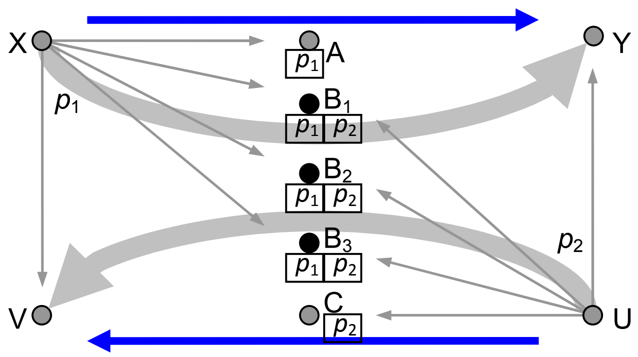


Figure 14 Neighborhood packet repository

various loci where two flows come close. Figure 14 depicts an example of such a local area of the network. In the figure, packet p_1 goes from node X via A to Y, and another packet p_2 goes from node U via C to V. These routes are determined by the routing protocol. In a multi-hop wireless network, some other nodes, say B_1, B_2, and B_3 can overhear the transmissions of p_1 and p_2. Traditional methods simply discard these overheard packets to avoid duplication, thus missing potential coding opportunities. Instead, BEND seizes the coding opportunities by enabling any one of B_1, B_2, or B_3 to forward $p_1 \oplus p_2$. This is a novel idea for exploiting the broadcasting nature of wireless channels. Once a packet, such as p_1 (p_2 , resp.) in this example, is transmitted, it is in effect received by all neighbors of the transmitter, i.e. nodes A, B_1, B_2 and B_3 (nodes B_1, B_2, B_3 and C, resp.). Instead of wastefully discarding the overheard packets, BEND stores them at the MAC layer and uses them later. In this way, these nodes share a significantly richer repository for coding by collectively snooping data communications in the neighborhood. BEND coordinates the coding and forwarding of the queued packets so that these nodes make use of such a repository jointly. The benefit is to enable more coding opportunities in the neighborhood, without forcing traffic flows through a fixed joint node, as required by COPE.

To use an analogy, a packet experiences such a proactive mixing of packets similar as the light photons experience the bending of a gravitational field. Here, the "gravity" for a packet arises from the possibility of combining it with other packets on potential forwarders en route. At each moment, the queues of the forwarders are likely to contain different packets due to the spatial diversity (e.g., forwarder positions), and the temporal diversity (e.g., traffic dynamics). Since a packet is likely to be stored at several forwarders, it tends to be forwarded by the one at which higher-gain coding (coded up with more packets) or any coding can be achieved. Due to the dynamics of the packets overheard and stored by nodes at different positions in the network, such tendency can change on a per-packet and per-hop basis. Therefore, a MAC-layer per-packet adaptation shows great applicability and flexibility in seizing coding opportunities. Globally, a packet's trajectory follows *only approximately* the exact route specified by the routing protocol. Each time a packet moves forward, it is transmitted by a node "around" the route determined by the routing module. For example in Figure 15, there is a flow between nodes *S* and *D* and its route is determined by the routing module as indicated by the thick light-colored line. Consider three back-to-back packets, p_1, p_2 and p_3. They can take different trajectories as shown in the figure to seize the coding opportunities in

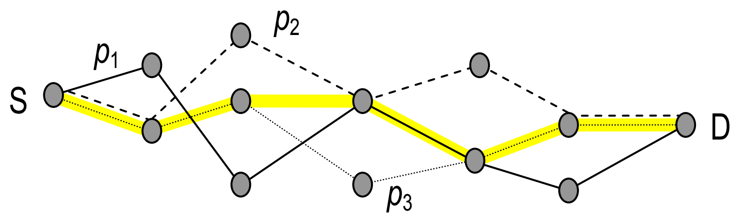


Figure 15 Closure of trajectories

different times and at different mixing nodes. This flexible and real-time adaptation offers high coding ratio and, hence, throughput gain. We believe such per-packet and per-hop decision-making is only feasible through a MAC layer implementation.

4.3 The design of BEND

4.3.1 Objectives and challenges

We design BEND with the following objectives:

BEND is based on IEEE 802.11 MAC [23] and should be easy to implement for practical use. Therefore, it should follow the 802.11 CSMA/CA paradigm and achieve reliable delivery for each transmission.

As a link layer solution, BEND should work with a routing protocol instead of making routing decisions. Indeed, choosing what type of routing protocol, proactive or not, source routing or distance vector, flat or hierarchical, position-based or energy-aware, is not necessarily only a performance issue, and should be left to the network operator. To that end, BEND should make minimum assumptions about the routing protocol used.

In order to design such an efficient mixing protocol, we must address the following challenges:

Maximizing coding chances — To promote coded transmissions for throughput gain, a mechanism is needed to ensure that packets have a better chance to be coded and transmitted by one forwarder than to be transmitted non-coded. This must be handled without starving any flows or nodes in the network.

Recognizing coding conditions — When a node has a packet to forward, it needs to know whether coding it with other queued packet(s) may save bandwidth; i.e. it needs to determine if the receivers can decode the coded packets.

Duplication of packets — All nodes operate in the promiscuous mode for opportunistic forwarding. As a result, a packet will be overheard and queued at multiple neighbors. There must be a mechanism to ensure that it is forwarded by only one of these neighbors.

Reliable link-layer broadcast — Since a coded packet is intended for multiple receivers, an efficient and reliable link-layer broadcasting mechanism is needed as a building block.

These challenges are addressed in the next section where details of BEND are presented.

4.3.2 Design details

In this section, we present the main components in the design of BEND. The basic

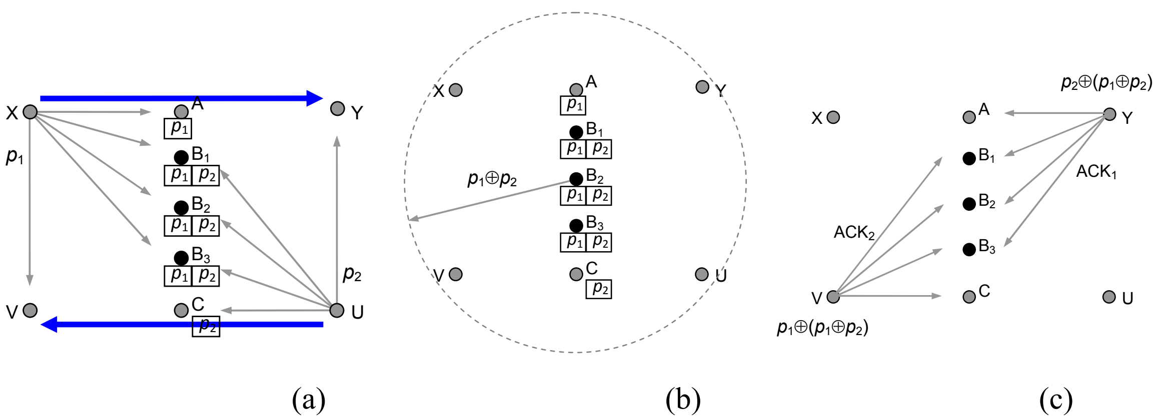


Figure 16 BEND – Design overview

operation of BEND is illustrated by a simple example in Figure 16 although BEND works under much more general conditions. In Figure 16(a), node X has packet p_1 for node Y that is two hops away, and node U has packet p_2 for node V, also two hops away. The forwarders determined by the routing protocol are nodes A and C, respectively. We further assume that three other nodes, B_1, B_2, and B_3, are also within the range of nodes X, Y, U, and V. When a packet, say p_1 or p_2, is handed from the network layer down to the MAC layer, its header is enhanced (Section 4.3.2.1 below) to include not only the address of the next-hop node but also that of the following-hop node. Such information can be obtained by querying the routing module (Section 4.3.2.2). After node X's packet p_1 and node U's packet p_2 are transmitted, p_1 is received by nodes A (intended forwarder), B_1, B_2, B_3 and V, and p_2 is received by nodes B_1, B_2, B_3, C (intended forwarder), and Y. For p_1, it is placed in the queues of nodes A, B_1, B_2, and B_3 because they are all neighbors of p_1's second-next-hop (node Y) as indicated by the packet header. Otherwise, it is buffered by node V for future decoding. Similarly, p_2 is queued at nodes B_1, B_2, B_3 and C, and buffered at node Y. Nodes B_1, B_2 and B_3 can choose to transmit $p_1 \oplus p_2$ if they determine that the coded packets can be correctly decoded by their second-next-hop neighbors (Section 4.3.2.3). All of the intermediate nodes A, B_1, B_2, B_3 and C could forward the packet(s) in their queues, coded or not. In order to expedite the packet forwarding, coded packets are transmitted with a higher priority, without starving uncoded packets (Section 4.3.2.4). Assume that node B_2 wins the channel and transmits $p_1 \oplus p_2$ (Figure 16(b)). The second-next-hop nodes V and Y receive the XORed packets and are able to decode them using the packets stored in their buffer. Then they immediately reply with an ACK in a "distributed bursty" fashion in the order specified by the enhanced MAC header. Such a reliable link-layer broadcast mechanism (Section 4.3.2.5) also helps to remove the packets queued at the intermediate nodes to avoid packet duplication (Figure 16(c)).

4.3.2.1 Header specification

BEND performs packet coding and tagging at a MAC sender. It requires a modification to the DATA and ACK headers of the existing 802.11 MAC Specifications [23].

In Figure 17, we show the header fields modified or added for BEND. The header of DATA frame may have a different format depending on whether the payload is encoded. If non-coded, in addition to the sender address (*SA*) and receiver address (*RA*), the header includes the IP address of 2nd-next-hop (described in Section 4.3.2.2 below). If encoded, it has a list *RA*[] of receiver addresses, and corresponding list *packetID*[] for all the encoded packets. The packet ID is generated by creating a 4-byte hash value out of the source's IP address and the sequence number carried by the IP packet, as in COPE. A 2-bit type and a 4-bit sub_type field in the frame-control field specify frame types, i.e., non-coded DATA, encoded DATA, ACK, NACK or other 802.11 frame.

Each ACK or NACK contains an *SA* and the packet ID of the original packet to

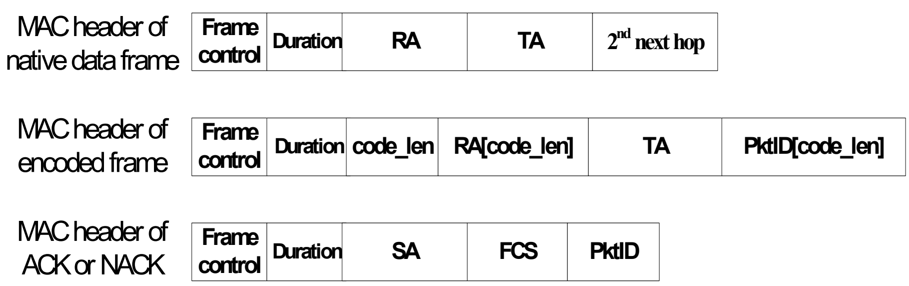


Figure 17 MAC headers of BEND

acknowledge. Notice that BEND uses *SA* in ACK instead of *RA* as in the 802.11 Specifications. The reason for this is described in Section 4.3.2.5 below.

4.3.2.2 2nd-next-hop en route

When a node requests help from its neighbors to forward a packet, it finds the IP address of the 2nd-next-hop (denoted by 2NH in the rest of the paper) along the path to the destination. If the destination is at least two hops away, it sets the 2nd-next-hop field in MAC header and transmits this DATA frame. This tells the potential forwarders where this packet should go next. The knowledge of 2NH is provided by the routing module. If a source or link-state routing is used, this is trivial. However, such knowledge is not immediately available for distance-vector based routing protocols. 2NH information can be obtained by minor modifications to distance-vector protocols. We simply add a “via” field to each distance vector in routing packets. That is, in the routing table broadcast to the neighbors, each entry destination is associated with distance estimation plus the neighbor via which this distance is established.

Upon receiving a DATA frame, only the nodes that are neighbors of the 2NH specified in the frame header are allowed to forward it. This guarantees that the packet propagation is restricted within a “stripe” along the route without flooding the network. When such a packet p_1 is received by a potential forwarder, it is passed up to the network layer with the 2NH information. The network layer fills in the next-hop field using the specified 2NH information and sends it down to the queue. BEND then searches the queue for mixing opportunities with other queued packets.

4.3.2.3 Queuing and mixing strategy

At each intermediate node, multiple packets can be mixed in a single transmission. For each pair of packets p_1 and p_2 in the set of packets to be combined, they must satisfy the following criteria:

1. The next-hop receiver of p_1 is p_2's previous forwarder, or one of its neighbors;

2. The next-hop receiver of p_2 is p_1's previous forwarder, or one of its neighbors.

The first condition ensures that the receiver of p_1 has p_2 in its buffer so that it can XOR p_2 with the coded packet. Likewise, the second condition ensures p_2 can be obtained by its corresponding receiver. These conditions are based on the assumption that a node's neighbors can receive its packets with a reasonable success ratio. Such link delivery ratio can be obtained by a network-layer routing metric, such as ETX [13]. For example, in Figure 16(a), p_1 and p_2 are queued at B_1. Here, p_1's next-hop receiver is Y, which happens to be a neighbor of p_2's previous forwarder U. Similarly, p_2's next hop is V, which is a neighbor of p_1's previous forwarder X. Thus, when p_1 and p_2 are encoded, to reconstruct p_1 node Y can XOR the coded packet with p_2, which was overheard from U earlier. Node V performs a similar operation to extract p_2. The probability that both Y and V can successfully decode the coded packet is a product of the delivery ratios of link XV and link UY. Therefore, as long as such decoding probability is higher than some threshold, we consider that the mixing criteria are met.

To realize the above conditions, we need to maintain at each node a 1-hop-neighbor and neighbor's-neighbor lists. These lists along with the link delivery information can be easily constructed and updated based on the routing protocol exchanges with the above “via” extension.

Each node stores packets that are intended for itself (with matching *RA*) and the packets that are overheard in different FIFO queues, denoted by Q_1 and Q_2, respectively. Those that satisfy the coding conditions are moved to a queue, denoted as *mixing-Q*. The packet matching process is as follows.

When a packet p_1 is passed down from the network layer, BEND searches the mixing-Q, Q_1 and Q_2 in this order for coding partners. BEND tries to mix as many packets as possible into a single coded transmission. The more packets are coded in one transmission, the higher throughput gain is achieved. Thus, it always starts the search with the mixing-Q. The condition of mixing more than two packets is that any two packets should satisfy the above pair-wise matching conditions. For example, suppose there are already two packets, p_2 and its coding partner p_3, in the mixing-Q. If pairs (p_1, p_2) and (p_1, p_3) further satisfy the matching criteria, we store p_1 in the mixing-Q along with p_2 and p_3. Otherwise, if no other partnerships, i.e. groups of two or more codable packets, can be found in the mixing-Q, BEND searches Q_1 and Q_2 in turn for 2-packet coding opportunities. It starts from the head of the queue, and the first matching packet will be removed and queued along with p_1 at the tail of the mixing-Q. If no matching can be found for p_1, it will be queued at the tail of Q_1 if this node is the intended forwarder, or Q_2 otherwise. The packets will be kept in queue for subsequent matching attempts until they are finally transmitted. If a packet is still alone when scheduled, it will be transmitted non-coded. Again, the forwarder will set its 2NH field so that other nodes down the path can code it.

4.3.2.4 Two-level prioritization

A packet and its copies could be queued up at different nodes (either the intended forwarder or potential forwarder helpers). The diversity among the forwarder nodes provides the packet various options to be combined with different numbers of and sets of packets, or not coded at all. To maximize coding opportunities, BEND gives coded transmissions higher priority in scheduling. This is achieved at two levels: within a node and among a set of contending nodes.

In a loaded network, end-to-end delay is dominated by queuing delays at individual nodes. Since the coding opportunity is transient, BEND is designed to seize these opportunities effectively. In a forwarding node, the mixing-Q is assigned a higher weight or priority than Q_1 and Q_2. The scheduler generates a random number uniformly between 0 and 1. If the number is > W_X and the mixing-Q is not empty, the node retrieves and combines a list of coding partners from the mixing-Q and schedules an encoded transmission. Otherwise, it schedules a non-coded packet. With these tunable weights W_X, BEND gives encoded packets better chances for transmission and yet does not starve the non-coded packets without coding opportunities.

When the forwarder nodes contend with each other to transmit their scheduled packets, BEND prioritizes them by assigning them different back-off durations before medium access based on the types of their packets. It is implemented through an EDCF-like type-specific mechanism of IEEE 802.11e [24]. The 802.11e EDCF regulates that, after a node decides to send a type of packet, it must back off for a fixed period (AIFS) and another time interval uniformly distributed between 0 and *cw*, where *cw* is a changing contention window size, to coordinate contending nodes. Initially, *cw* is set to

CW_{Min} and is doubled every time a transmission attempt fails until it reaches the specified CW_{Max}. The smaller CW_{Min}, CW_{Max} and AIFS are, the higher priority is given to one type of packet. As in Table 1, we assign a higher access priority to transmissions that could achieve higher coding gain. The prioritization is important in BEND to achieve high throughput for two reasons. First, it coordinates potential forwarders' accesses to make best use of the coding repository. The nodes with more efficient combination can capture the media with higher possibilities. Second, the proactive mixing and forwarding in BEND may incur more medium access attempts in the area and thus more intense contention. Transmission classification and prioritization are necessary to effectively alleviate such contention and reduce the number of collisions.

Table 1 Parameters for packet prioritization

Type	CW_{Min}	CW_{Max}	AIFS
Overheard non-coded	99	2047	7 × slot time + SIFS
Intended non-coded	63	1023	4 × slot time + SIFS
2-packet coded	41	1023	3 × slot time + SIFS
3-packet coded	23	63	2 × slot time + SIFS
x-packet coded ($x > 3$)	9	63	2 × slot time + SIFS

We use the specific priority settings as in Table 1 and set W_X to 0.2 in all our experiments. However, the priority settings can be finer-tuned and determined by other important factors, such as delay requirement of traffic, or size and content of the queues. For example, if the historical statistics suggest no coding opportunities, the medium access delay for the non-coded packets can be lowered for better performance. The optimization of these parameters and its impact in different scenarios is left to future research.

When packets for mixing are scheduled, they are coded by XOR and the result is encapsulated with a MAC header for encoded frames (Figure 17). The number of packets encoded is specified in the *code_len* field. Their packet IDs and corresponding receiver addresses are also attached in the fields *PktId*[] and *RA*[] in the header. Then, the forwarder transmits this coded packet and waits for replies from the receivers.

4.3.2.5 Decoding, acknowledgement and retransmission

When a coded packet arrives at a receiver, the receiver checks whether its MAC address is in the *RA*[] list in the header. If so, it uses the positions of the other receivers to get the IDs of their packets from the *PktId*[] list. These packets are retrieved from this receiver's buffer and used to extract the packet intended for this receiver. These stored packets had either been forwarded or originated by this node before, or they had been overheard by this node when transmitted as non-coded over the medium. Again, the mixing strategy of Section 4.3.2.3 ensures that this node was in the neighborhood of the transmitters so it can hear them with reasonable probabilities. All the packets for decoding are stored in an FIFO buffer. If all packets for decoding are found, the node then decodes its non-coded packet and proceeds to send an ACK. Otherwise, it returns a NACK.

Since a coded packet is broadcast to multiple receivers, the link layer is responsible for the reliability of the broadcast. The 802.11 Specification only includes an unacknowledged, and thus unreliable, broadcast mechanism. Prior work, e.g. COPE, resorts to an approximate reliability. Here, we devise a reliable link-layer broadcast. In essence, all receivers of a coded packet are polled by the sender in the order specified in the *RA*[] field of the coded DATA header. So, the receivers send their ACKs back-to-back to the sender without collision.

In addition to reliable link transmission, another important task of ACK is to avoid packet duplication. In BEND, an ACK is used to free all copies of the delivered packet at the previous forwarders. To do that, the ACK contains the MAC address of the ACK sender instead of the receiver as in the regular 802.11 and the ID of the received non-coded data packet. When a node receives the ACK, it searches for the corresponding packet in the queue using packet ID in the ACK frame. If the ACK's sender (*SA*) is the next-hop node of the data packet, which means that the packet has already been successfully received by its next-hop receiver, this packet can be removed to avoid duplication.

The forwarder of the coded transmission will retransmit the NACKed or non-responded (timeout) packets. For the NACKed, it has to be retransmitted non-coded since it cannot be decoded with current combination. If there are no replies at all from any receivers, it is very likely that there was a collision. Thus, it increases its back-off time based on Table 1 and transmits the same coded packet again.

Once a packet is successfully decoded by an intermediate node, the node can either mix it with other packets in the queue, if there are any coding opportunities, or forward it as a plain packet to next-hop potential forwarders/mixers. This cycle repeats until the packet is delivered to the destination. Therefore, a packet could possibly be coded several times on different intermediate nodes with other packets from various flows. With such implementation, BEND promotes network coding among the traffics in the network.

4.4 Performance Evaluation

We use ns-2, a packet-level simulator, as a basis to test BEND's performance in various scenarios and compare it with IEEE 802.11 and COPE-Sim (an ns-2 implementation of COPE), to find how effective they are in supporting multiple flows in multi-hop wireless networks. We measure the aggregate throughput gain of BEND over COPE-Sim and over 802.11, and investigate how this gain is achieved through other measures and what can affect such a gain.

Our PHY layer model adopts BER (bit error rate) to introduce random packet loss to simulate more realistic operation conditions. Here, we use a BER of 2×10^{-6} so that, after an interface has received a packet, even if its strength exceeded the reception threshold, it may still be dropped with a probability. We fix the data rate at 1Mbps, the basic rate, without any rate adaptation, although any other fixed data rate would not change the relative performance among the protocols under test. With the two-ray propagation model in ns-2, the transmission range in this case is 250m. The data flows in the network are all CBR flows of 1000-byte datagrams and with an arrival interval of 0.01s and duration of 100s. In general, in each tested scenario, the combination of the flows saturates the network to test the protocols' maximum transportation capabilities. The network uses DSDV to determine routes between sources and destinations.

We test a set of scenarios with different characteristics to investigate BEND relative to COPE-Sim and 802.11. We start with a 3-tier scenario to test the coding capability with multiple flow pairs. Next, we use a cross topology to observe how BEND and COPE can seize the chances of coding 3 or 4 packets in a single transmission. Then we generalize to a 5×5 mesh topology with randomly deployed flows to investigate the effect of hop length and number of flows on these three protocols.

4.4.1 3-tier topology

In a 3-tier network, tiers 1 and 3 each consist of 4 nodes, and tier 2 may contain 1, 2, 3, or 4 nodes, referred to as 4-1-4, 4-2-4, 4-3-4, 4-4-4 topologies, respectively. We set the separation distance between tiers to 200m so that flows between tiers 1 and 3 must use tier-2 node(s) as forwarders. The distance among nodes of the same tier is small. In each of the four topology variants, we place 4 CBR flows randomly between tiers 1 and 3, two in the forward (left-to-right) direction and two in the reverse (right-to-left) direction. Since there are more routes between a source and a destination when we increase the number of tier-2 nodes, the chance that a forward flow and a reverse flow cross at a common forwarder decreases. On the other hand, when more forwarders are available in tier-2, this stage will become less of a bottleneck because, if the flows are routed via different nodes of tier-2, these forwarders collectively will have a better chance to capture the wireless channel than the case when all flows must be routed through a single forwarder as in 4-1-4. Hence, a higher diffusion gain is achieved.

We first measure the aggregate throughput that sums the number of packets arrived at the four UDP receiving agents. The plot in Figure 18(a) compares the throughputs of BEND, COPE-Sim and 802.11 in the four topology variants. As seen, 802.11 achieves a higher throughput when increasing the number of tier-2 nodes thanks to the higher diffusion gain due to more forwarders. For COPE-Sim, when it enjoys higher diffusion gain introduced by additional tier-2 nodes, it loses its coding power due to load scattering. BEND, however immediately utilizes the maximum benefit since adding the second forwarder. The throughput gains of BEND and COPE-Sim over 802.11 are plotted in Figure 18(b). Here, we define the throughput gain of a protocol over 802.11 as the ratio of the throughput of the protocol to that of 802.11 minus 1. For the 4-1-4 topology, where all flows go through the single tier-2 node, both COPE-Sim and BEND can almost double the network throughput by applying network coding at this forwarder. In contrast, when there are at least 2 nodes in tier-2 to provide alternative paths, BEND (55%~97%) offers nearly double throughput gain over 802.11 compared to COPE-Sim (29%~51%). This consistently higher gain of BEND is realized by allowing tier-2 nodes to transmit more coded packets even if the flows do not necessarily cross at a single node as in the 4-1-4 configuration. To verify this, we record the coding ratio, the number of packets forwarded as coded to the total number of packets forwarded by the tier-2 nodes, for the four topology variants (Figure 18(c)). For COPE-Sim, the coding ratio is lost by about two thirds (from 94% to 38%) as the focal nodes vanish among the forwarders; but BEND manages to lose just slightly over one third (from 94% to 57%). Note that error bars in Figure 18 indicate that, among the repeated simulation runs, BEND has much smaller variances than COPE-Sim and 802.11 in throughput and coding ratio. That is, BEND's performance is not affected by the specific routes determined by the routing module, while COPE-Sim and 802.11 are very sensitive to the level of flow concentration caused by routing. With BEND, diffusion gain and coding gain, which are otherwise conflicting factors, are unified by its power of proactive packet mixing.

BEND solves COPE's dilemma of simultaneously achieving coding gain and diffusion gain. For COPE-Sim running in presence of multiple tier-2 nodes, either concentrating flows at a particular node provides the coding gain or scattering flows among the forwarders provides the diffusion gain, but *not* both. For example, we take a set of 200 pairs of simulation of 802.11 and COPE-Sim over the 4-3-4 network, each pair records the performance of 802.11 and COPE-Sim using the same routes determined by DSDV. We sort them in the increasing order of the numbers of coded transmissions of

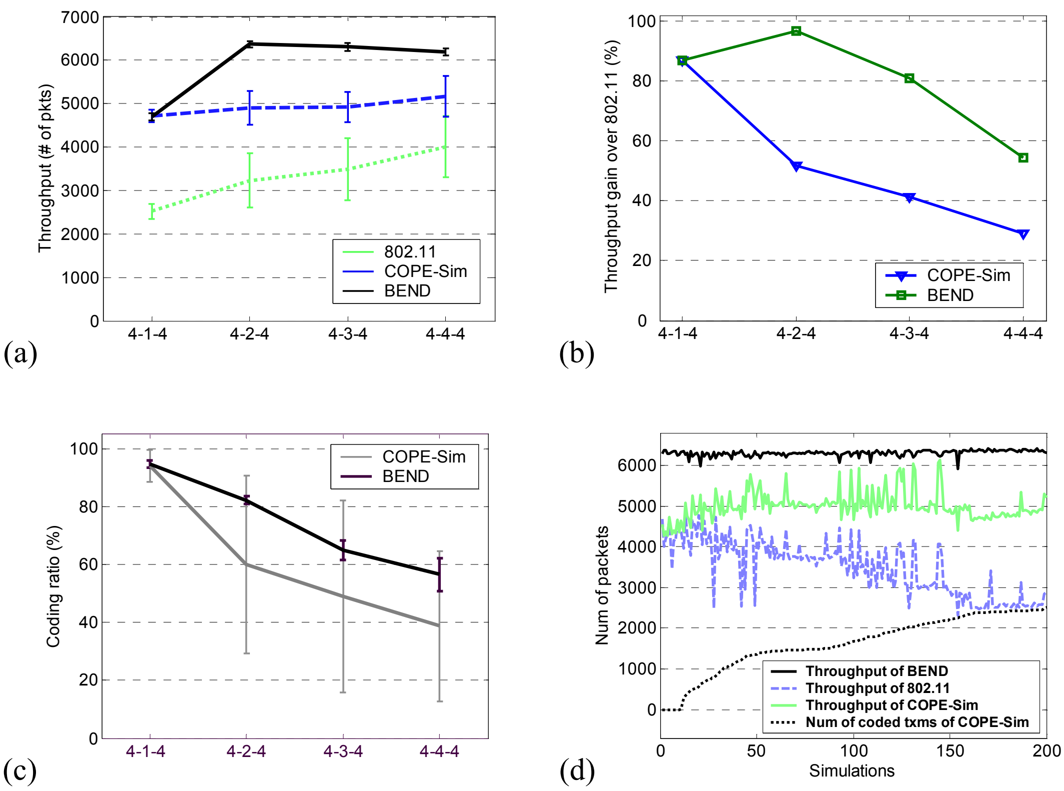


Figure 18 (a) Throughput of different methods. (b) Throughput gain over 802.11. (c) Coding ratios. (d) Negative correlations between coding gain and diffusion gain.

COPE-Sim. Then we display them with the throughputs of 802.11, COPE-Sim and BEND in Figure 18(d). Clearly, there is a negative correlation of 802.11's throughput and the number of coded transmissions of COPE-Sim. To the left, where no two reverse flows cross the same tier-2 node, 802.11 achieves its highest throughput but COPE cannot code a single pair of packets. To the right, where all flows cross through a common tier-2 node, 802.11 encounters severe bottleneck effects but COPE-Sim can transmit most packets coded. For BEND, the three tier-2 nodes work as an entity by processing the neighborhood coding repository among themselves, showing a persistently higher throughput gain over COPE-Sim.

4.4.2 Cross topology

To investigate the capability of coding more than two packets of BEND and COPE-Sim, we design a cross topology of radius 150m. As a result of the ns-2 settings, each peripheral node has three neighbors (the center and two other orthogonal peripheral nodes) and the center node has four (the peripheral nodes). We place four CBR flows originating from each of the peripheral nodes and terminating at the opposite node.

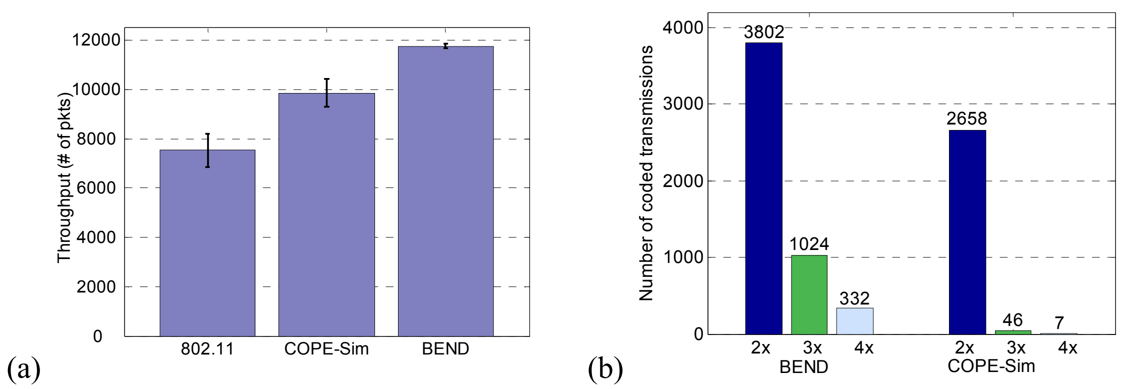


Figure 19 (a) Throughput; (b) Coded transmissions.

DSDV in this case can prescribe three paths for each flow. In such a configuration, up to four packets can be coded in a single transmission of the center node and up to two packets can be coded together by a peripheral node.

Figure 19(a) is the aggregate throughput of the flows supported by 802.11, COPE-Sim and BEND. Again, BEND achieves about double throughput gain relative to 802.11 (56%) compared to COPE-Sim relative to 802.11 (30%), with a much more stable performance.

We have also made the histogram of the number of 2-packet codings, 3-packet codings, and 4-packet codings for BEND and COPE-Sim (Figure 19(b)). Apparently, the chance that COPE-Sim is able to code more than two packets is very slim given that DSDV will often route more than three flows through a same forwarder. Considering that the four peripheral nodes can only code 2 packets during a single forwarding, the coding of 3 or 4 packets indicates that BEND is very effective in seizing coding opportunities. This is achieved without introducing any artificial backlogging or delay at forwarders although, apparently, holding packets or even pairs or triplets of coded packets in the queue for a bit longer can further boost these numbers. However, we choose not to use this to avoid any delay just for the sake of coding.

4.4.3 Mesh topology

We further generalize to a 5×5 mesh topology to test the performance of BEND in supporting random flows in larger networks. In the configuration, the grid distances in the two orthogonal directions are set to 150m. Thus, a non-peripheral node has eight neighbors and a corner node has three neighbors. The diameter of the network is four

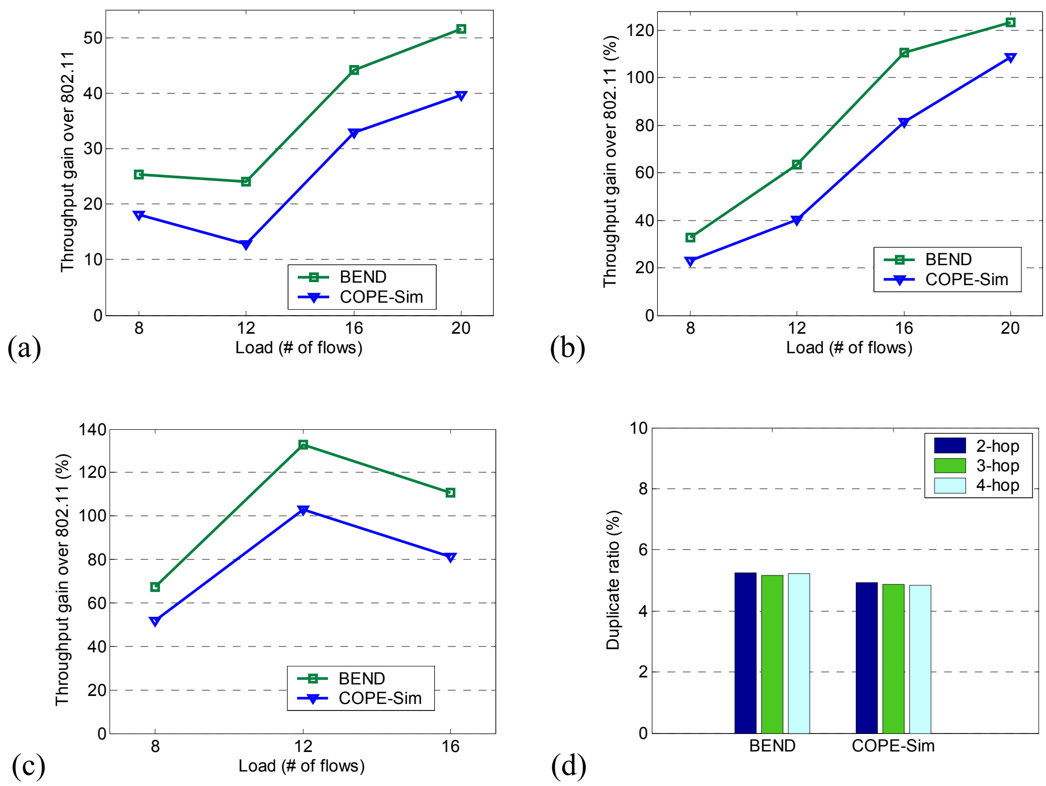


Figure 20 (a) 2-hop flows; (b) 3-hop flows; (c) 4-hop flows (d) duplicate ratio

hops. It is known that multi-hop flows take considerably more network resources to transport the same amount of data. Our goal in this set of experiments is to study BEND's effectiveness with regard to COPE-Sim and 802.11 in supporting a varying number of flows with differing lengths.

We individually test the cases of different flow lengths originating from distinct nodes in the network. For the case of transporting l-hop (l = 2, 3, and 4) flows, we vary the number of flows in the network among f = 8, 12, 16, and 20. Note that the combination of l = 4 and f = 20 is impossible given the network diameter of 4 and only 16 peripheral nodes. We plot the throughput of BEND, COPE-Sim and 802.11 for a given hop length in a chart (Figure 20(a), (b) and (c), respectively). We notice that BEND consistently offers a higher throughput gain than COPE-Sim. In addition, a general trend is that when the number of flows increases, more coding opportunities are found for both BEND and COPE-Sim.

We are also interested in, in a larger network, how effective the reliable broadcast and duplication mechanisms are. To measure this, we record the rate of duplicate packets received at UDP receivers for both protocols for each hop-length case averaged over all flow numbers tested. As shown in Figure 20(d), the duplication rate of BEND is marginally higher than that of COPE-Sim.

4.5 Discussion

Traditional routing protocols' obliviousness to the coding opportunities was noticed in [44][48][54]. Their solutions focus on routing at the network layer. Such attempts are usually referred to as coding-aware routing. The idea is to compute routes for given flows in a network, taking network coding gain into account, so that the expected total number of transmissions needed to transport the flows is minimized. This is of great importance in theory but the distributed implementation can be rather involved. To compute coding-efficient paths, each node needs to maintain global information of all the flows in the network. The time granularities of traffic lifetime and route update period are usually discrepant. The calculated routes will typically be long dated before being applied to the flows used for the route calculation. More so, due to the extremely close coupling among these flows, any unilateral change of route adopted by an intermediate node will invalidate the purpose of the global routing metric, which is to reduce the number of transmissions. Moreover, coding-aware routing approach is still based on traditional routing with a single fixed path for each source-destination pair and the redundancy of packets in the network cannot be utilized.

BEND aims at achieving a high coding ratio for each stage of forwarding. It is not globally optimal, but it is flexible, adaptive and practically effective. It only requires local information and the implementation overhead is low. Since it is a per-packet decision for coding, as opposed to per-flow path adaptation, it is more responsive to the dynamics of traffic. BEND also takes advantage of packet redundancy in the network by opportunistic forwarding. The coding chances are greatly improved with multiple potential forwarders instead of one. Moreover, coding-aware routing needs to consider not only the coding gain by combining traffic flows but also their consequential interference. These two forces have been difficult to balance with traditional methods of fixed-path routing. By allowing a set of nodes to process the neighborhood packet repository collectively, BEND adapts to the flow dynamics in the network. On one hand, it proactively mixes data flows that would otherwise go through different nodes in the neighborhood as specified by the routing module. On the other hand, when a specific node happens to be a junction of multiple routes and becomes overloaded, BEND diffuses flows in its neighborhood to alleviate the bottleneck effect. These two aspects are unified under the same framework of BEND. The idea of BEND resembles that of ExOR [8] and MORE [10]. In ExOR, any neighbor en route can forward an overheard data packet as long as it determines that such an opportunistic forwarding leads the packet closer to its destination. Unlike ExOR, which prioritizes forwarders by their distances to the destination, BEND favors those forwarders with a chance to transmit coded packets. ExOR reduces the number of transmissions along the path by skipping some hops if by chance they are received by a node closer to destination. MORE enhances ExOR with network coding to further reduce the transmission redundancy in delivering a single flow from source to destination, i.e. *intra-flow coding*. In contrast, BEND accomplishes efficient delivery by finding more *inter-flow coding* opportunities in the network.

BEND requires no more routing information than what a distance vector protocol normally offers. In a multi-hop wireless network, the discrepancy among the fragments of routing information maintained at individual nodes of the network can affect the coding decision of BEND to a degree. As a result, there can be occasions where a coded packet cannot be decoded at a receiver because the sender mistakenly decided that decoding was possible using some inconsistent routing information. Even with such minimum requirement for the routing protocol used, BEND can well tolerate such errors and achieve higher data transportation capabilities as shown in the experiments. We believe that the performance of BEND can be further improved if a link-state or a source routing protocol is used where a much more consistent global routing map is stored at each node. However, this is a significantly stronger assumption about the routing module that will compromise the compatibility of BEND with routing protocols.

4.6 Conclusion and Future Work

Broadcasting can cause interference in multi-hop wireless networks, but it also brings the benefit of facilitating network coding. When applied effectively, network coding will significantly improve the network's transportation capabilities. The BEND protocol proposed in this chapter starts off with the goal of creating more network coding opportunities with a low overhead. It averts the impasse of possibly scarce coding opportunities as with COPE. The key of BEND is to create more coding chances via proactive traffic mixing by treating the packets queued at a neighborhood of forwarders collectively as a distributed packet repository. Our simulation studies indicate that, with minimum assumption on the routing module, BEND consistently achieves higher throughput support than without proactive traffic mixing as in COPE.

Current implementation of BEND makes use of packet redundancy for the encoding aspect. That is, any intermediate node in the neighborhood can encode and forward packets. BEND can be further extended to use packet redundancy for the decoding aspect. In this case, after a node receives a coded packet, if it cannot decode for the non-coded packet intended for it due to some earlier transmission errors, any of its neighbors could decode the packet alternatively and pass the non-coded packet further on to the next hop. One difficulty in realizing this for a distance-vector routing protocol is that this may necessitate the acquisition of the "third-next-hop neighbor" information and including it in the packet header. If a link-state or source routing protocol was adopted instead, this would be a relatively easy extension but would impose a stronger assumption on the routing information. It is also possible to further increase the coding ratio of BEND by introducing more sophisticated delay and scheduling in packets as in [11]. This can be another avenue to fine-tune BEND. There are some other challenges about making coding decisions. For example, the coding gain becomes marginal when a large-size packet and a small packet are encoded together. In addition, when the link qualities between the sender and the multiple next-hop receivers of the coded transmission are different, the sender has to use the most conservative transmission rate to make sure every receiver can successfully receive the packet. From above, all these factors have to be considered in coding decision-making toward high throughput gain.

Chapter 5

Non-Intrusive 802.11 Link Quality Estimation

Compared to wired link, wireless communication suffers from high transmission error ratio and its performance changes frequently and dramatically, highly depending on the channel quality, which usually exhibits great variability. The sources of variation include user mobility, environmental changes and interference. The rapidly varying channel condition results in changing packet error rate (PER), therefore making network bandwidth a highly dynamic resource. Due to the dynamic variability, the wireless link quality needs to be measured and provided to wireless applications and protocols, in order for wireless networks to most effectively utilize and manage resources.

In a wireless mesh network, the knowledge of the wireless link quality can help applications and protocols to adapt their behaviors to improve their own performances and network efficiency. For example, QoS-sensitive applications need wireless link bandwidth information to perform admission control and resource allocation. Another important application is path finding in WMN. As mentioned in section 2.5, a routing metric considering the characteristics of each link is more effective than the minimal-hop metric, which ignores the variability presented in wireless links.

This chapter presents a novel approach to estimate 802.11 link bandwidth based on radio signal-to-noise ratio (SNR). Our objective is to monitor the wireless link as it appears on top of the MAC layer. *Wireless link bandwidth* is defined as the effective transmission bandwidth of a wireless link or *saturated throughput* that can be achieved at the wireless link. It is different from the nominal or “ideal” channel bandwidth, which do not account for error rate and lengthy retransmission time in the MAC layer.

We exploit the wireless broadcasting feature to achieve non-intrusive estimation. That is, any transmissions overheard by a node can be used to evaluate the link bandwidth between this node and the corresponding transmitters. We show the theoretical analysis and experimental observation of the relationship between SNR and wireless link bandwidth. Then, we provide two modeling methods for two different modulation rates in IEEE 802.11b respectively.

5.1 Background

There exist a number of methods for route bandwidth estimation in wired networks, such as pathchar [14] and packet-pair [45][35]. However, these methods are intrusive to the network since they introduce overhead traffic by sending probe packets during the estimation process. The overhead is not desirable in wireless networks where network bandwidth is usually scarce and precious. In addition, due to the dynamics of the wireless link, there is a need for more frequent bandwidth measurement, thus consuming even more resources than in wired networks. Therefore, a non-intrusive bandwidth estimation method is required for wireless networks.

The existing work on non-intrusive or passive measurement in wireless networks, e.g., [32][34][49], estimates the wireless bandwidth by observing the RTTs of payload packets of other applications running on the same host instead of sending own probe packets. However, these methods are inaccurate because they cannot control the size and transmission time of the packets. E.g., [34] , estimates the bandwidth based the time interval between packet pair arrivals, which may be inaccurate for small packets due to the coarse granularity of the system clock, thus degrading the estimation performance.

The method in [31] performs bandwidth estimation on a per-packet basis at the MAC layer. For each packet, the transmission and ACK reception times are recorded and used in estimation. The method factors in the collision avoidance delay in the presence of multiple senders and thus provides the permissible throughput. However, it requires MAC protocol support and the accuracy depends on the packet size.

There is a great amount of work on channel quality estimation at physical and link layers, which employs sophisticated electronics or sending pilot symbols. Unfortunately, it would require modifications to commonly available 802.11 NIC adapter cards.

Traditionally, signal-to-noise ratio (SNR) or carrier-to-interference ratio (CIR) is measured and reported as an indicator of wireless link quality. Most of the 802.11 WLAN cards measure and display the signal strength (usually as signal bars). There is a prior literature [22][46] on adapting network parameters, e.g., link transmission rate, based on SNR, which assumes high correlation between SNR and link quality. The theoretical relationship between the SNR and link quality, represented by bit error rate (BER), can be derived for an additive white Gaussian noise (AWGN) channel. The SNR and BER relationships for different 802.11b modulation techniques are shown in Figure 21(a). The link bandwidth is the maximal throughput that can be achieved over the MAC layer. It can be determined from BER, as in Figure 21(b), for packets of a fixed length (adjusted for the overhead of PHY and MAC layers). In one of our datasets collected for DBPSK, shown in Figure 21(c), the link bandwidth values are distributed around the theoretical DBPSK curve.

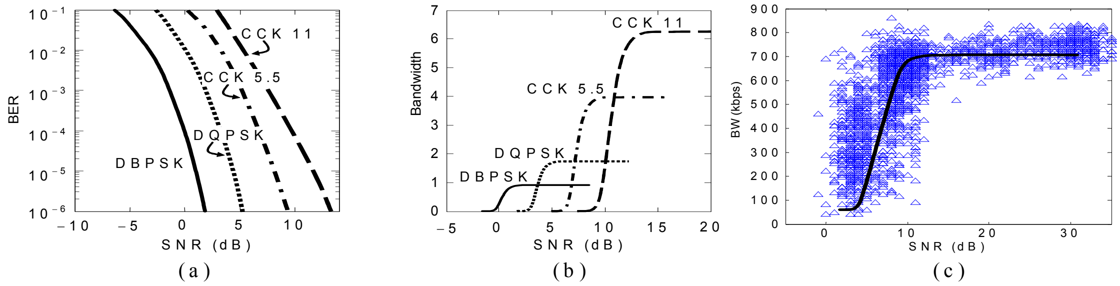


Figure 21 SNR vs. bandwidth relationship for AWGN channel. (a) Q-functions for different modulation schemes. (b) Corresponding SNR-BW theoretical relationships. (c) Measured SNR-BW for DBPSK.

Our first method [63], Single-point mapping, on estimating the 802.11 link bandwidth from measured SNR is based on the above theoretical support and experimental observation.

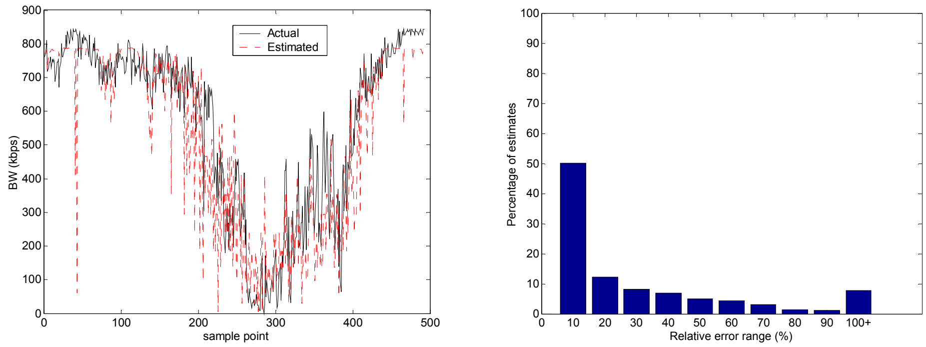
5.2 The single-point mapping

In theory, there is a well defined relationship between the SNR and the wireless link bandwidth, denoted as *B(t)*:

$$B(t) \xleftrightarrow{\eta[\cdot]} PER(t) \xleftrightarrow{P[\cdot]} BER(t) \xleftrightarrow{Q[\cdot]} SNR(t) \tag{5}$$

where BER and PER are bit error rate and packet error rate, respectively. Functions η, P and Q define the theoretical relationships.

However, in reality, it is very difficult to build the exact theoretical model above. For this reason, below we consider empirical methods for bandwidth estimation. Nonetheless, the above analysis is important to back our intuition about the existence of the relationship.



(a) (b)

Figure 22 Performance of BPNN model: (a) Comparison of measured and estimated bandwidths; (b) The distribution of relative errors

The method consists of three components: data collection, modeling, and estimation. To build and test the model, we need to collect sufficient amount of data of SNR and link bandwidth. We ignore the issues of multi-user medium sharing on the bandwidth measurement since here we only deal with the channel quality. Therefore, a single transmitter is connected to a single receiver in 802.11 ad-hoc mode and both are located in different offices. The modulation rate is fixed for each round of measurement. A UDP flow is started with the rate high enough to saturate the link. The sampling period of the series is 1 second. The receiver records an SNR point for each received frame. To obtain the SNR over 1-second intervals, we average the SNR's of 20 randomly selected frames that were received during that second. If less than 20 frames were received in a second, all of them are used. For each 1-second interval, the link bandwidth is also calculated as the product of the frame size and the number of the received frames.

The data collected as above are used to build the model, i.e., training. When building a model for recorded datasets, we try to build a model such that the relationship of its output (estimated bandwidth) to its input (SNR) matches what is exhibited by most of the points in the datasets. When the model is used in estimation, real-time measured

SNRs are fed into the model and the generated outputs are estimates of bandwidth. In order to evaluate the accuracy of the model, the estimated values are compared to the actual bandwidth, which is again measured by sending probe packets. We use absolute mean error and average relative error as indices for performance evaluation. The average relative error is defined as $\left(\sum|y_i - \hat{y}_i|\right)/\sum y_i$, where y_i and $\hat{y}_i$ are the desired and estimated outputs at i^{th} point, respectively.

We first train a 1-4-1 Back-propagation neural network (BPNN) on a certain dataset and then it is used to generate estimates by feeding other datasets of recorded SNRs. The estimates on each dataset are compared with the corresponding actual measured link bandwidth, and errors are presented in the form of relative and absolute mean value. The relative error between the estimated bandwidth and the actual measured bandwidth in Figure 22(a) is 14.81%. Figure 22(b) shows the distribution of estimates in different relative error ranges. We may notice that for about 50% of estimates relative error is less than 10%. We repeated testing the BPNN model by training it on 5 datasets and applied it to do estimation on 5 different datasets. The average relative error obtained is 24.94% and the standard deviation of error is 13.57%.

5.3 Time-series modeling

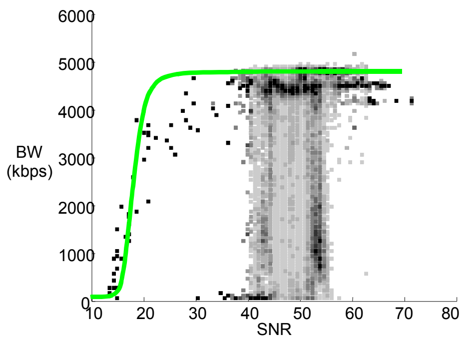


Figure 23 Distribution of a dataset on BW-SNR space.

The previous method is based on the assumption of AWGN channel. However, the above assumption of a non-fading AWGN channel can be invalid in reality, especially in typical office and home environments with fading links. The data collected in MIT Roofnet show inconsistency of the relationship between PER and SNR or transmitter-receiver distance. They conclude that the observed large number of links with moderate error rates is probably due to multipath fading rather than signal attenuation. In [36], the authors claim that, although an average error rate curve for their multipath scenario has been obtained by averaging over a large number of randomly generated channels, most of their observed PER is significantly lower than this average curve. Similar situation is observed in our experimental results, especially those collected in office environments with high transmission rate (CCK 11) which tends to be more sensitive to multipath. In Figure 23, the measured throughput values are distributed in the gray zone ranging from about 0 up to 5Mbps, although SNR's are all above 40dB, which by the non-fading curve should yield high link bandwidths.

Therefore, individual SNR points alone do not adequately describe the wireless channel quality for 802.11 links with fading channels, i.e., the two-dimensional scatter plot curve is not sufficient to model their relationship. Realizing the limitations of a simple correlation, [36] suggests a substitute indicator for the prediction of PER, the computation of which, however, requires ideal channel estimation in PHY. Here, we propose a time-series modeling method, assuming that SNR time series may provide more information on link quality than single points of SNR. Instead of a single-point SNR, the current and historical SNR's and corresponding bandwidths are treated as two time-series signals and their relationship is modeled as a time series transfer function. Like the previous method, the model is identified and fitted to the SNR signal and corresponding link bandwidth in a training dataset. This model is tested on other data, collected in the same environment, to determine if there exists regularity in the relationship between SNR and link bandwidth. Then, we can use the model to predict the link bandwidth in similar scenarios.

We test two time-series modeling techniques, Auto-Regressive Moving Average eXogenous variables (ARMAX) model, and Recurrent Neural Network (RNN) technique, more specifically. The former is linear, and the latter nonlinear.

5.3.1 ARMAX Modeling

In a single-input, single-output linear causal ARMAX model [53], the relationship between output series $\boldsymbol{y_t}$ and the input series $\boldsymbol{x_t}$ can be described through a linear filter as:

$$y_t = v_0 x_t + v_1 x_{t-1} + v_2 x_{t-2} + \cdots + n_t \qquad (6)$$

where $\boldsymbol{n_t}$ is a noise series of the system that is independent of the input series $\boldsymbol{x_t}$. Eq. (6). can be rewritten as:

$$y_t = v(B) \cdot x_t + n_t \tag{7}$$

where $v(B) = \sum_{j=0}^{\infty} v_j B^j$ is called the transfer function of the filter, and $\boldsymbol{B}$ is the backward shift operator. The goal of the modeling is to identify and estimate the transfer function $\boldsymbol{v(B)}$ and the noise $\boldsymbol{n_t}$ based on the available information of the input series $\boldsymbol{x_t}$ and the output series $\boldsymbol{y_t}$. In our case, these correspond to the SNR and the link bandwidth, respectively. As seen in the equations, the current bandwidth output of the ARMAX model is not only a function of the current SNR point, but also depends on the current and past SNR points.

To build an ARMAX model, the input series $\boldsymbol{x_t}$ is assumed to follow some ARMA($\boldsymbol{p}$, $\boldsymbol{q}$) model, i.e., $\boldsymbol{x_t}$ satisfies

$$\phi_x(B)x_t = \theta_x(B)\alpha_t \tag{8}$$

where $\boldsymbol{\alpha_t}$ is a white noise series, and $\boldsymbol{\phi}(\cdot)$ and $\boldsymbol{\theta}(\cdot)$ are the $\boldsymbol{p}$th and $\boldsymbol{q}$th degree polynomials as

$$\phi_x(B) = 1 - \phi_1 B - \cdots - \phi_p B^p \tag{9}$$

$$\theta_x(B) = 1 - \theta_1 B - \cdots - \theta_q B^q \tag{10}$$

We compute the sample autocorrelation function (ACF) and the sample partial autocorrelation function (PACF) of the preprocessed series to identify the orders of $\boldsymbol{p}$ and $\boldsymbol{q}$. By comparing the Akaike information criterion (AIC) of the models with different combinations of $\boldsymbol{p}$ and $\boldsymbol{q}$, we find that the best fitted model is an ARMA(2,6) model, like so

$$(1-0.131B-0.541B^2)x_t = (1-0.603B-0.566B^2 + 0.189B^3 + 0.034B^4 + 0.053B^5 - 0.050B^6)\alpha_t \tag{11}$$

The variance of the noise $\boldsymbol{\alpha_t}$ is 0.0126.

Due to the nonlinearity in the signals, the linear ARMAX model shows high error on the prediction output. In addition, the input SNR series and the output link-bandwidth series have to be transformed to stationary series for the ARMAX modeling. The output has to be postprocessed to reverse the preprocessing operations, which demands some unavailable prior knowledge in prediction process, such as the initial value of the output series for reversing the differencing. Considering the above inadequacies of the ARMAX modeling, we turn to nonlinear time-series modeling. The Recurrent Neural Network (RNN) technique, more specifically Echo State Network (ESN) , is chosen due to its ability for nonlinear modeling and prediction. The ESN combines various linear and nonlinear operations and automatically tunes the corresponding parameters based on the training input/output series. The preprocessing is simple, as is the postprocessing.

5.3.2 ESN modeling

ESN is an efficient black-box modeling method for nonlinear predication [25][26]. The idea is to construct a model with a series of linear operations (weighting and summation), nonlinear operations, and delay operations such that it mimics a given empirical dataset.

An ESN is a network of artificial neurons. A neuron is a basic computational unit that computes some function, usually nonlinear, of the weighted sum of inputs from other units or an external source. Its output, in turn, can be served as input to other units. ESN has both feed-forward and feedback connections. For example, in Figure 24, the output or activation of unit 1 is updated according to

$$x_1(t) = f\big(w_1 \cdot x_1(t-1) + w_2 \cdot x_2(t) + w_3 \cdot x_3(t)\big) \tag{12}$$

where $\boldsymbol{w_1}$, $\boldsymbol{w_2}$, $\boldsymbol{w_3}$ are weights assigned to the three inputs of unit 1. Its inputs, $\boldsymbol{x_1(t-1)}$, $\boldsymbol{x_2(t)}$, and $\boldsymbol{x_3(t)}$, are delayed output feedback from itself, current output of unit 2 and current output of unit 3, respectively. The output of internal units is called state. In Figure 24, the input signals are introduced by the input layer to the internal layer. The internal

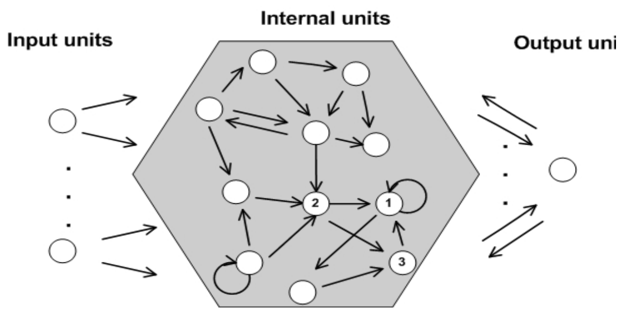


Figure 24 Structure of an echo state network

units update their states at each time step as in Eq. (12). The output of the ESN is then decided by the states as follows

$$y(t) = f^{out}\left(\sum_{i=1}^{N} w_i^{out} \cdot x_i(t)\right) \tag{13}$$

Eqs. (18) and (19) decide the relationship between the input SNR and the output bandwidth in our ESN model. They can be viewed as re-expression and association, respectively, similar to power transformation and fitting in ARMAX. By Eq. (12), the input signal is transformed or re-expressed, by the internal neurons, as the states which expose principal patterns hidden in the input series. This mechanism provides richer nonlinear expression than the power transformation. By the ESN training algorithm, the output weights in Eq. (13) are updated automatically so that the revealed patterns are associated with the desired output. This is done by adjusting the output weights $\boldsymbol{w_i^{out}}$ so that the error $\boldsymbol{e(n)}$ in Eq. (14) is minimized, in the mean square error sense, i.e., the difference between the output of the model and the desired output is minimized

$$e(t) = (f^{out})^{-1}\left(y_{desired}(t)\right) - (f^{out})^{-1}\left(\sum_{i=1}^{N} w_i^{out} \cdot x_i(t)\right) \tag{14}$$

where $\boldsymbol{(f^{out})^{-1}(\cdot)}$ is inverse function of $\boldsymbol{f^{out}}(\cdot)$, and $\boldsymbol{y_{desired}}$ is the desired output. After training, ESN can start doing prediction when supplied by a real-time SNR input signal.

We create an ESN with 400 internal units, single output (bandwidth), and single input (SNR), and select 5000-point training dataset. The ESN model is trained and the tuning process is repeated until the mean square error on the training data reaches desired low level. After that, we check our model by predicting the link bandwidth from the measured SNR in a different dataset with a total of 20,000 points. (Recall that both the training and estimation datasets are collected in a similar environment). The results are shown in Figure 25. The output of ESN is the estimated link bandwidth, which is the dashed brighter-colored curve in the figure. The actual link bandwidth is the solid curve. The bottom row in Figure 25(a) shows smoothed estimated bandwidth which may be more suitable for use by adaptive applications. The figure shows good agreement between the estimated and the actual values. Figure 25(b) shows the distribution of the estimates in different relative error ranges. It shows that around 30% of points are estimated with relative error lower than 10%. For the same dataset, the ESN model outperforms both the two-dimensional non-fading model and the ARMAX model. The non-fading model fails to predict the degradation of link bandwidth due to the fading since it always overestimates the bandwidth, given the high SNR input. Compared to ARMAX, the ESN model provides variety of options for re-expression of SNR series, which contributes to the performance. Despite the overall estimate accuracy improvement, on the other hand, we also notice there are around 20% of points that are estimated with high error (over 100%). The high relative error happens mainly at the points of extremely low actual bandwidth, which is an artifact of how we calculate the relative error. Another reason for those high error points could be the incompleteness of the training dataset. That is, some input/output sequences are not present in the training dataset. When they appear in the estimating dataset, their corresponding bandwidth cannot be estimated correctly. However, the bigger the training dataset is, the longer and

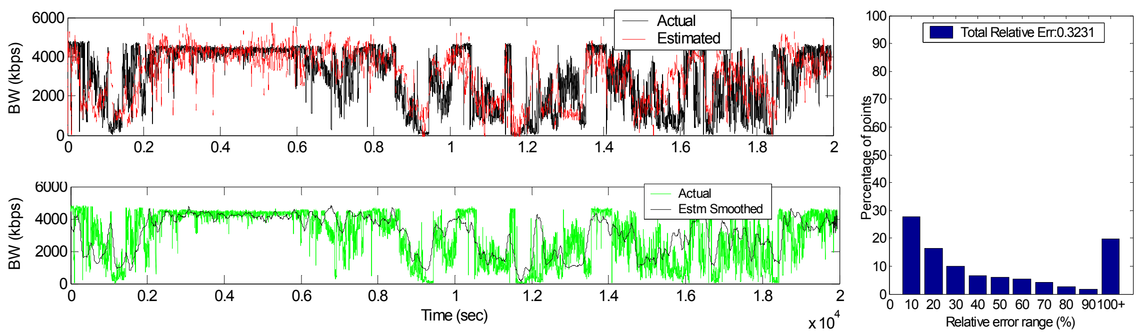


Figure 25 (a) Comparison of actual and estimated bandwidth and estimation error by ESN model. The bottom row shows the estimated bandwidth smoothed by averaging over a window of 100 points; (b) Relative estimation error.

costlier the computation. Finding a complete, yet non-redundant training dataset is still an open problem.

5.4 Conclusions

In this chapter, we study the relationship of SNR and bandwidth in both fading and non-fading environments by single-point modeling and time-series modeling, respectively. Our experimental data confirm that individual SNR points do not adequately describe the wireless link quality with multipath and fading. Instead, by the time-series modeling, the patterns of SNR sequences can be recognized and associated with the corresponding link quality. The process of building a linear ARMAX model indicates nonlinearity in the SNR series. The nonlinear ESN model, with its ability of providing rich re-expressions and associations of signals, achieves accurate prediction on link bandwidth. This shows that even under multipath fading, the relationship of the SNR and link bandwidth can be captured by a combination of linear, nonlinear, and delay operations.

Chapter 6 Conclusions and Future Work

The relay functionality of wireless mesh networking provides extended coverage and cost-effectiveness for wireless networks. However, these advantages come at a price. As multi-hop wireless networks, wireless mesh networks, pose a challenge in network protocol design.

The two of the most critical aspects in wireless mesh networking are the error-prone communication links and unstable network structure. Numerous efforts have been exerted to address these issues so that a multi-hop wireless network is as good as wireline networks. Oppositely, there has been increasing interests in utilizing wireless communication channels by harnessing its unique nature of broadcasting. Indeed, it is this nature that separates wireless networks from the rest, and there is no need to turn wireless links into wired lines. Only by exploiting this nature can we make full use of wireless networks and improve their performances greatly.

The benefit from the nature of broadcasting is two-fold. First, broadcasting leads to redundancy, thus robustness. A single transmission of a packet may leave multiple copies of the packet on different neighbor stations of the transmitter. Such redundancy is important and should be exploited, especially to counteract the effect of wireless link errors. That is, when the intended station fails to decipher the packet, other stations can take over the tasks with the overheard copy of the packet. Second, there is diversity in the process of broadcasting. At each moment, mesh stations in a local area may differ from each other with their physical locations, device specs, channel conditions (e.g., interference), MAC-layer statuses (e.g., idle or busy), reception results of the same transmission, packets overheard in the local area, and so on. Each of these can be utilized for designing protocols in WMNs.

The main contribution of this dissertation is to exploit the broadcasting feature for high throughput performance in WMNs. To achieve this goal, we have described three techniques which take advantage of wireless broadcasting in different aspects.

- **MAC-status-based scheduling (**Chapter 3**)** We have developed a solution to head-of-line blocking problem in WMNs. We make a sender probe the MAC-layer availability of multiple receivers. With historical observation of the probes, the diversity among their MAC-layer statuses can be speculated. Such diversity further helps the sender reschedule the packets and deliver them in higher efficiency.
- **Network coding with opportunistic forwarding** (Chapter 4) We propose a new protocol called BEND, which enables each potential forwarder to proactively mix/encode the packets that either are intended to or are overheard by this node. In BEND, a richer packet repository for coding is constructed and maintained by each station in that local area. Therefore, the coding ratio is significantly improved.
- **Wireless link quality estimation (**Chapter 5) We have presented a method to model and estimate 802.11 link bandwidth based on radio signal-to-noise ratio (SNR). This method is a non-intrusive in that the estimation can be made on overheard signals.

When designing and implementing the prototype of the above solutions, we have tried to make them compatible with the existing prevailing protocol, IEEE 802.11. In fact, our proposed methods can easily be extended for the mesh networks deployed with different technologies, such as IEEE 802.16.

The future work about how to enhance each of the three techniques is given separately at the end of the corresponding chapters. Some other research directions for future work include: (a) Relay coordination using wireless broadcasting and diversity. Such coordination can be achieved in PHY layer, such as MIMO, in MAC-/Network-layer, such as the methods in this dissertation, or a combination, i.e., a cross-layer solution. (b) Capacity modeling when relay coordination is enabled.

References

[1] F. B. Abdesslem, L. Iannone, M. D. de Amorim, K. Kabassanov & S. Fdida, "On the feasibility of power control in current IEEE 802.11 devices," in *Proceedings of the 4th Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops 2006)*

[2] D. Aguayo, J. Bicket, S. Biswas, G. Judd & R. Morris, "Link-level measurements from an 802.11b mesh network," in *Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM '04)*, Vol. 34, No. 4, pp 121-132, October, 2004.

[3] R. Ahlswede, N. Cai, S. Y. R. Li & R. W. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, Vol. 46, No. 4, pp 1204-1216, 2000.

[4] I. F. Akyildiz, X. Wang & W. Wang, "Wireless mesh networks: a survey," *Computer Networks*, Vol. 47, No. 4, pp 445-487, March 2005.

[5] B. Awerbuch, D. Holmer & H. Rubens, "The medium time metric: high throughput route selection in multi-rate ad hoc wireless networks," *Mobile Networks and Applications*, Vol. 11, No. 2, pp 253-266, April, 2006.

[6] P. Bhagwat, P. P. Bhattacharya, A. Krishna & S. K. Tripathi, "Enhancing Throughput over Wireless LANs Using Channel State Dependent Packet Scheduling," in *Proceedings of the 15th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '96)*, pp 1133-1140, 1996.

[7] V. Bharghavan, A. Demers, S. Shenker & L. Zhang, "MACAW: a media access protocol for wireless LAN's," in *Proceedings of the conference on Communications architectures, protocols and applications (SIGCOMM '94)*, pp 212-225, 1994.

[8] S. Biswas & R. Morris, "ExOR: opportunistic multi-hop routing for wireless networks," in *Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM '05),* pp 133-144, October, 2005.

[9] M. Cesana, D. Maniezzo, P. Bergamo & M. Gerla, "Interference aware (IA) MAC: an enhancement to IEEE802.11b DCF," in *Proceedings of the IEEE 58th Vehicular Technology Conference (VTC 2003-Fall)*, Vol. 5, pp 2799-2803, 2003.

[10] S. Chachulski, M. Jennings, S. Katti & D. Katabi, "Trading structure for randomness in wireless opportunistic routing," *SIGCOMM Comput. Commun. Rev.*, Vol. 37, No. 4, pp 169-180, October, 2007.

[11] P. Chaporkar and A. Proutiere, “Adaptive network coding and scheduling for maximizing throughput in wireless networks,” in *Proceedings ACM MobiCom*, Montreal, Canada, 2007

[12] C. C. Chen, E. Seo, H. Kim & H. Luo, “Self-Learning Collision Avoidance for Wireless Networks,” in *Proceedings of the 25th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '06)*, pp 1-12, 2006.

[13] D. De Couto, D. Aguayo, J. Bicket & R. Morris, “A High-Throughput Path Metric for Multi-Hop Wireless Routing,” in *Proceedings of the 9th ACM International Conference on Mobile Computing and Networking (MobiCom'03)*, 2003.

[14] A. B. Downey, “Using pathchar to estimate Internet link characteristics,” in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (ACM SIGCOMM’99)*, Vol. 29, No. 4, pp 241-250, October, 1999.

[15] R. Draves, J. Padhye & B. Zill, “Comparison of routing metrics for static multi-hop wireless networks,” in *Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM '04)*, pp 133-144, 2004.

[16] R. Draves, J. Padhye & B. Zill, “Routing in multi-radio, multi-hop wireless mesh networks,” in *Proceedings of the 10th annual international conference on Mobile computing and networking (MobiCom '04)*, pp 114-128, 2004.

[17] J. Eriksson, S. Agarwal, P. Bahl & J. Padhye, “Feasibility study of mesh networks for all-wireless offices,” in *Proceedings of the 4th international conference on Mobile systems, applications and services (MobiSys 2006)*, pp 69-82, 2006.

[18] C. Fragouli, V. Sivaraman & M. B. Srivastava, “Controlled Multimedia Wireless Link Sharing via Enhanced Class-Based Queueing with Channel-State-Dependent Packet Scheduling,” in *Proceedings of the Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '98)*, pp 572-580, 1998.

[19] J. Gomez, A. T. Campbell, M. Naghshineh & C. Bisdikian, “Conserving transmission power in wireless ad hoc networks,” in *Proceedings of the 9th International Conference on Network Protocols*, pp 24-34, 2001.

[20] A. A. Hamra, C. Barakat & T. Turletti, “Network coding for wireless mesh networks: a case study,” in *Proceedings of the International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2006)*, pp 9, 2006.

[21] R. Hekmat & P. Van Mieghem, “Interference in wireless multi-hop ad-hoc networks and its effect on network capacity,” *Med-hoc-Net*, 2002.

[22] G. Holland, N. Vaidya & P. Bahl, “A rate-adaptive MAC protocol for multi-Hop wireless networks,” in *Proceedings of the 7th annual international conference on Mobile computing and networking (MobiCom '01)*, pp 236-251, 2001.

[23] IEEE, “Wireless LAN medium access control (MAC) and physical layer (PHY) specifications,” *IEEE standard 802.11*, 1997.

[24] IEEE 802.11 WG, Draft Supplement to Standard for Telecommunications and Information Exchange between Systems-LAN/MAN Specific Requirements- Part 11: Wireless LANMAC and Physical Layer (PHY) Specifications: Medium Access Control (MAC), Enhancements for QoS, *802.11e Draft 4.1*, February 2003.

[25] H. Jaeger, “The ‘echo state’ approach to analysing and training recurrent neural networks,” *GMD Report 148, German National Research Center for Information Technology*, 2001.

[26] H. Jaeger, H. Haas, “Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communications,” *Science*, 78-80, April 2, 2004.

[27] S. Jain & S. R. Das, “Exploiting Path Diversity in the Link Layer in Wireless Ad Hoc Networks,” in *Proceedings of the Sixth IEEE International Symposium on World of Wireless Mobile and Multimedia Networks (WOWMOM '05)*, pp 22-30, Washington, DC, USA, 2005.

[28] E. S. Jung & N. Vaidya, “A Power Control MAC Protocol for Ad Hoc Networks,” *Wireless Networks*, Vol. 11, No. 1-2, pp 55-66, January, 2005.

[29] S. Katti, S. Gollakota & D. Katabi, “Embracing wireless interference: analog network coding,” in *Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM '07)*, pp 397-408, 2007.

[30] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard& J. Crowcroft, “XORs in the air: practical wireless network coding,” in *Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM '06)*, pp 243-254, 2006.

[31] M. Kazantzidis & M. Gerla, “End-to-end versus Explicit Feedback Measurement in 802.11 Networks,” in *Proceedings of the Seventh International Symposium on Computers and Communications (ISCC'02)*, 2002.

[32] M. Kim & B. Noble, “Mobile network estimation,” in *Proceedings of the 7th ACM Conf. Mobile Computing and Networking (MobiCom'01)*, pp 298-309, 2001.

[33] T. S. Kim, J. C. Hou & H. Lim, “Improving spatial reuse through tuning transmit power, carrier sense threshold, and data rate in multihop wireless networks,” in *Proceedings of the 12th annual international conference on Mobile computing and networking (MobiCom '06)*, pp 366-377, 2006.

[34] K. Lai & M. Baker, “Nettimer: A Tool for Measuring Bottleneck Link Bandwidth,” in *Proceedings of the Third USENIX Symposium on Internet Technologies and Systems*, pp 123-134, March 2001.

[35] K. Lai & M. Baker, “Measuring Bandwidth,” in *Proceedings of the 18th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '99)*, pp 235-245, 1999.

[36] M. Lampe, H. Rohling & W. Zirwas, “Misunderstandings about link adaptation for frequency selective fading channels,” in Proceedings *of the 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Vol. 2, pp 710-714, 2002.

[37] M. J. Lee, J. Zheng, Y.-B. Ko & D. M. Shrestha, “Emerging standards for wireless mesh technology,” *IEEE Wireless Communications*, Vol. 13, No. 2, pp 56-63, 2006.

[38] H. Lim, C. Lim & J. C. Hou, “A coordinate-based approach for exploiting temporal-spatial diversity in wireless mesh networks,” in *Proceedings of the 12th annual international conference on Mobile computing and networking (MobiCom '06)*, pp 14-25, 2006.

[39] D. S. Lun, M. Medard & R. Koetter, “Network Coding for Efficient Wireless Unicast,” in *Proceedings of International Zurich Seminar on Communications*, pp 74-77, 2006.

[40] S. A. Mahmud, S. Khan, S. Khan & H. Al-Raweshidy, “A comparison of MANETs and WMNs: commercial feasibility of community wireless networks and MANETs,” in *Proceedings of the 1st International Conference on Access Networks (AcessNets '06)*, USA, 2006.

[41] J. P. Monks, V. Bharghavan & W. M. W. Hwu, “A power controlled multiple access protocol for wireless packet networks,” in *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '01)*, Vol. 1, pp 219-228, 2001.

[42] A. Muqattash & M. Krunz, "POWMAC: a single-channel power-control protocol for throughput enhancement in wireless ad hoc networks," *IEEE Journal on Selected Areas in Communications*, Vol. 23, No. 5, pp 1067-1084, 2005.

[43] V. Navda, S. Das & S. Ganguly, "Interference-aware fast path adaptation in wireless mesh networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, Vol. 11, No. 2, pp 63-64, 2007.

[44] B. Ni, N. Santhapuri, Z. Zhong & S. Nelakuditi, "Routing with opportunistically coded exchanges in wireless mesh networks," in *Proceedings of the 2nd IEEE Workshop on Wireless Mesh Networks (WiMesh 2006)*, pp 157-159, 2006.

[45] V. Paxson, "End-to-end Internet packet dynamics," *IEEE/ACM Transactions on Networking*, Vol. 7, No. 3, pp 277-292, 1999.

[46] D. Qiao & S. Choi, "Goodput enhancement of IEEE 802.11a wireless LAN via link adaptation," in *Proceedings of IEEE International Conference on Communications (ICC 2001)*, Vol. 7, pp 1995-2000, 2001.

[47] C. Reis, R. Mahajan, M. Rodrig, D. Wetherall & J. Zahorjan, "Measurement-based models of delivery and interference in static wireless networks," *SIGCOMM Comput. Commun. Rev.*, Vol. 36, No. 4, pp 51-62, October, 2006.

[48] S. Sengupta, S. Rayanchu & S. Banerjee, "An Analysis of Wireless Network Coding for Unicast Sessions: The Case for Coding-Aware Routing," in *Proceedings of the 26th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2007)*, pp 1028-1036, 2007.

[49] S. H. Shah, Kai Chen & K. Nahrstedt, "Available Bandwidth Estimation in IEEE 802.11-based Wireless Networks," in *Proceedings of 1st ISMA/CAIDA Workshop on Bandwidth Estimation*, 2003

[50] Y. C. Shalinee, "Wireless Diversity through Network Coding," in *Proceeding of IEEE Wireless Communications and Networking Conference (WCNC'06)*, 2006

[51] F. Tobagi & L. Kleinrock, "Packet Switching in Radio Channels: Part II--The Hidden Terminal Problem in Carrier Sense Multiple-Access and the Busy-Tone Solution," *IEEE Transactions on Communications*, Vol. 23, No. 12, pp 1417-1433, 1975.

[52] J. Wang, H. Zhai & Y. Fang, "Opportunistic packet Scheduling and Media Access control for wireless LANs and multi-hop ad hoc networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '04)*, Vol. 2, pp 1234-1239, 2004.

[53] W. S. Wei, “Time Series Analysis: Univariate and Multivariate Methods”, Addison Wesley Publ. Co., Reading, MA, 1990.

[54] Y. Wu, S. Das & R. Chandra, “Routing with a Markovian Metric to Promote Local Mixing,” *Microsoft Research Technical Report, MSR-TR-2006-158*, November 2006

[55] S. H. Y. Wong, S. Lu, H. Yang & V. Bharghavan, “Robust rate adaptation for 802.11 wireless networks,” in *Proceedings of the 12th annual international conference on Mobile computing and networking (MobiCom '06)*, pp 146-157, 2006.

[56] X. Yang & N. Vaidya, “On physical carrier sensing in wireless ad hoc networks,” in *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2005)*, Vol. 4, pp 2525-2535, 2005.

[57] H. Yomo & P. Popovski, “Opportunistic Scheduling for Wireless Network Coding,” in *Proceedings of the IEEE International Conference on Communications (ICC '07)*, pp 5610-5615, 2007.

[58] J. Zhang, Y. P. Chen & I. Marsic, “Adaptive MAC Scheduling Using Channel State Diversity for Wireless Networks,” in *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM 2006)*, pp 1-4, 2006.

[59] J. Zhang, Y. P. Chen & I. Marsic, “MAC Scheduling Using Channel State Diversity for High-Throughput IEEE802.11 Mesh Networks,” *IEEE Communications Magazine*, Nov 2007.

[60] J. Zhang, Y. P. Chen & I. Marsic, “Persistent Pseudo-Clearance Problem in IEEE802.11 Mesh Networks and its Multicast Based Solutions,” in *Proceddings of the 15th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN’07)*, Princeton, June, 2007.

[61] J. Zhang, Y. P. Chen & I. Marsic, “Network Coding Via Opportunistic Forwarding in Wireless Mesh Networks,” in *IEEE Wireless Communications & Networking Conference* (WCNC), 2008.

[62] J. Zhang, Y. P. Chen & I. Marsic, “BEND: MAC-Layer Proactive Mixing Protocol for Network Coding in Multi-Hop Wireless Networks,” in *the Nineth ACM International Symposium on Mobile Ad Hoc Networking and Computing* (*MobiHoc*) *Posters*, 2008.

[63] J. Zhang, L. Cheng & I. Marsic, “Models for non-intrusive estimation of wireless link bandwidth,” in *Proceedings of Personal Wireless Communications, IFIP-TC6 8th International Conference (PWC 2003)*, 2003.

[64] J. Zhang & I. Marsic, “Link Quality and Signal-to-Noise Ratio in 802.11 WLAN with Fading: A Time-Series Analysis,” in *Proceedings of the IEEE 64th Vehicular Technology Conference*, Montreal, Canada, Sept 2006.

[65] J. Zhu, X. Guo, L. L. Yang & W. S. Conner, “Leveraging spatial reuse in 802.11 mesh networks with enhanced physical carrier sensing,” in *Proceedings of the IEEE International Conference on Communications (ICC '04)*, Vol. 7, pp 4004-4011, 2004.

[66] The list of US cities and counties with wireless networks, URL: http://www.muniwireless.com/article/articleview/5495

[67] Champaign-Urbana Community Wireless Network, URL: http://www.cuwireless.net/

[68] MIT roofnet, URL: http://pdos.csail.mit.edu/roofnet/

[69] SeattleWireless, URL: http://www.seattlewireless.net/

[70] Wireless mesh network by Microsoft Research, URL: http://research.microsoft.com/mesh/

Curriculum Vitae

Jian Zhang

9/1990 – 7/1994	Huazhong University of Sci & Tech, Wuhan, P. R. China. Bachelor of Engineering in Control Engineering.
9/1994 – 7/1997	Huazhong University of Sci & Tech, Wuhan, P. R. China. Master of Science in Control Theory.
9/1998 – 6/2000	Software Engineer, Huateng Software Systems Co., Shanghai, P.R. China.
1/2002 – 10/2008	Rutgers, The State University of New Jersey, Department of Electrical and Computer Engineering, New Brunswick, New Jersey. Doctor of Philosophy in Electrical and Computer Engineering.

Publications

Book chapter and journal papers

1. J. Zhang, Y. P. Chen and I. Marsic, MAC Scheduling Using Channel State Diversity for High-Throughput IEEE802.11 Mesh Networks, *IEEE Communications Magazine*, Nov 2007.
2. Y. P. Chen, J. Zhang and I. Marsic, Enhancing Efficiency and Effectiveness of 802.11 MAC in Wireless Mesh Networks, *Handbook of Wireless Mesh and Sensor Networking*, McGraw-Hill, 2008.

Conference papers

3. J. Zhang, Y. P. Chen and I. Marsic, BEND: MAC-Layer Proactive Mixing Protocol for Network Coding in Multi-Hop Wireless Networks, *ACM MobiHoc'08 posters*.
4. J. Zhang, Y. P. Chen and I. Marsic, Network Coding Via Opportunistic Forwarding in Wireless Mesh Networks, *Proceedings of IEEE WCNC'08*, Apr, 2008.
5. Y. P. Chen, J. Zhang and A. N. Ngugi, An Efficient Rate-Adaptive MAC for IEEE802.11, *Proceedings of MSN'07*, 2007.
6. J. Zhang, Y. P. Chen and I. Marsic, Persistent Pseudo-Clearance Problem in IEEE802.11 Mesh Networks and its Multicast Based Solutions, *Proceedings of LANMAN'07*, Princeton, June, 2007.

7. J. Zhang, Y. P. Chen and I. Marsic, Adaptive MAC Scheduling Using Channel State Diversity for Wireless Networks, *Proceedings of WiCOM'06*, Wuhan, China, Sept 2006.
8. J. Zhang and I. Marsic, Link Quality and Signal-to-Noise Ratio in 802.11 WLAN with Fading: A Time-Series Analysis, *Proceedings of IEEE VTC'06*, Montreal, Sept 2006.
9. Y. P. Chen, D. Wang and J. Zhang, Variable-Base Tacit Communication: A New Energy Efficient Communication Scheme for Sensor Networks, *Proceedings of InterSense'06*, France, 2006.
10. J. Zhang, L. Cheng and I. Marsic, Models for Non-intrusive Estimation of Wireless Link Bandwidth, *Proceedings of the IFIP-TC6 PWC'03*, Venice, Italy, Sept 2003.